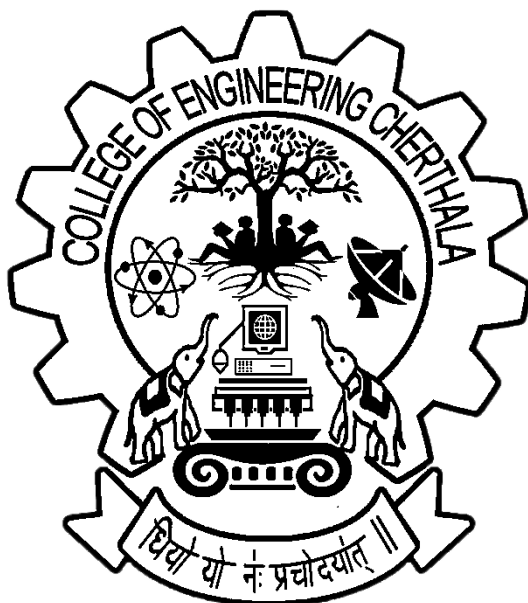


COLLEGE OF ENGINEERING CHERTHALA

LAB RECORD

20MCA133 – WEB PROGRAMMING LAB



CERTIFICATE

*This is certified to be bonafide works of Mr./Ms.
....., In the class, Reg. No.,
of College of Engineering Cherthala, during the academic year 2022-23.*

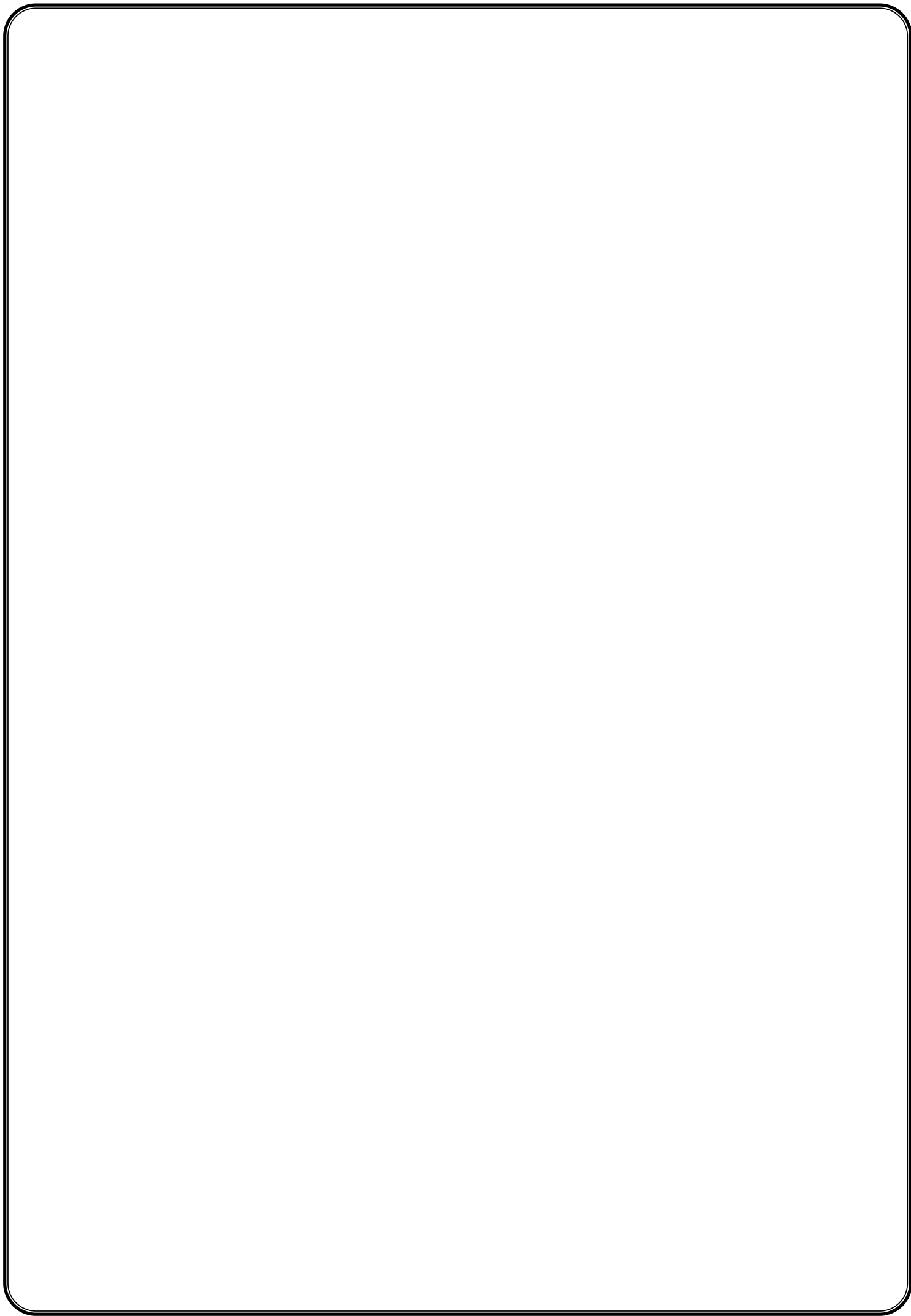
Teacher In Charge

External Examiner

Internal Examiner

INDEX

Ex. No.	Name of Experiment	Page No.	Date of Experiment	Remarks
1	Demonstration of different HTML tags	01	19/10/2022	
2	Link different HTML pages	03	26/10/2022	
3	HTML page with different frames	06	02/11/2022	
4	HTML files using different styles	08	09/11/2022	
5	Registration form using HTML	10	16/11/2022	
6	Predefined functions in a string and math object in javascript	14	23/11/2022	
7	Calender using Javascript	21	7/12/2022	
8	Form Validation	25	14/12/2022	
9	Background change using JavaScript	28	21/12/2022	
10	JavaScript Event Handling	30	04/01/2023	



Program No.1

Demonstration of different HTML tags

Aim :- To create a simple HTML file to demonstrate the use of different tags

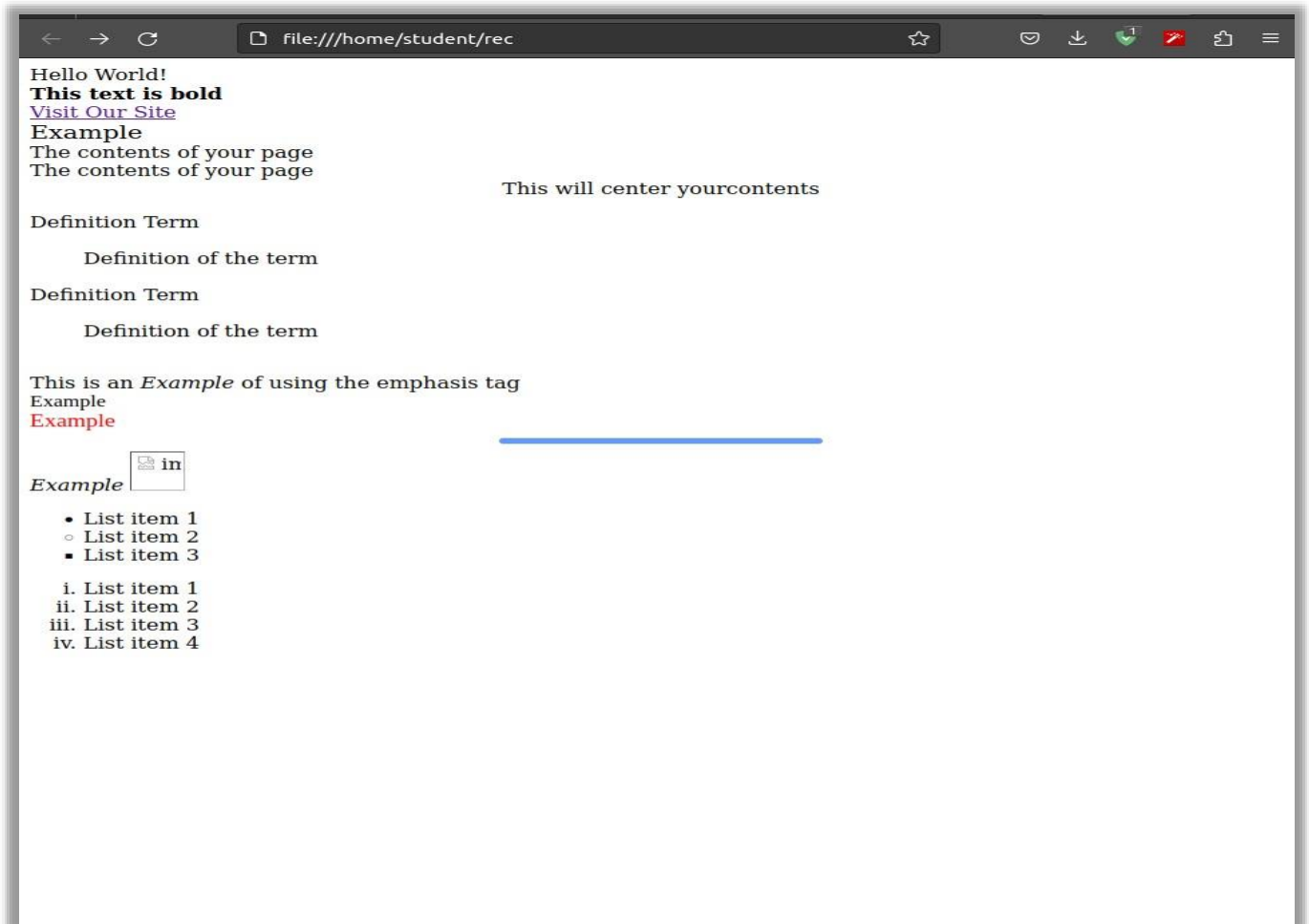
Algorithm :-

1. Open a text editor or HTML editor.
2. Start with an HTML5 doctype declaration: <!DOCTYPE html>.
3. Add the opening and closing html tags.
4. Add the head section within the html tags.
5. Within the head section, add a title tag to give the page a title.
6. Add a link tag within the head section to link to a stylesheet if needed.
7. Close the head section.
8. Add the body section within the html tags.
9. Add a header section within the body tags to include any header information you want.
10. Add a nav section within the body tags to create a navigation menu for the links.
11. Within the nav section, create a list of links using the ul and li tags.
12. Use the a tag within each li tag to link to the appropriate HTML pages.
13. Add a section tag within the body tags to contain the main content of the page.
14. Within the section tag, create any additional div or article tags needed to organize the content.
15. Add an img tag within the section tag to include any images needed.
16. Use the table tag to create any tables needed within the section tag.
17. Within the section tag, use the a tag to create links within the same page using href="#anchor" and id="anchor" to specify the location of the link.
18. Add a footer section within the body tags to include any footer information you want.
19. Close the body section and the html section.

Program :-

```
<html>
<head>
<title>HTML_tags</title>
</head>
<body>
Hello World!
<b>This text is bold</b>
<!-- This is a comment -->
<a href="http://www.google.com/">Visit Our Site</a>
<big>Example</big> <!-- big -->
The contents of your page<br>The contents of your page
<center>This will center yourcontents</center>
<dl>
<dt>Definition Term</dt>
<dd>Definition of the term</dd>
<dt>Definition Term</dt>
<dd>Definition of the term</dd>
</dl>
```

Output :-

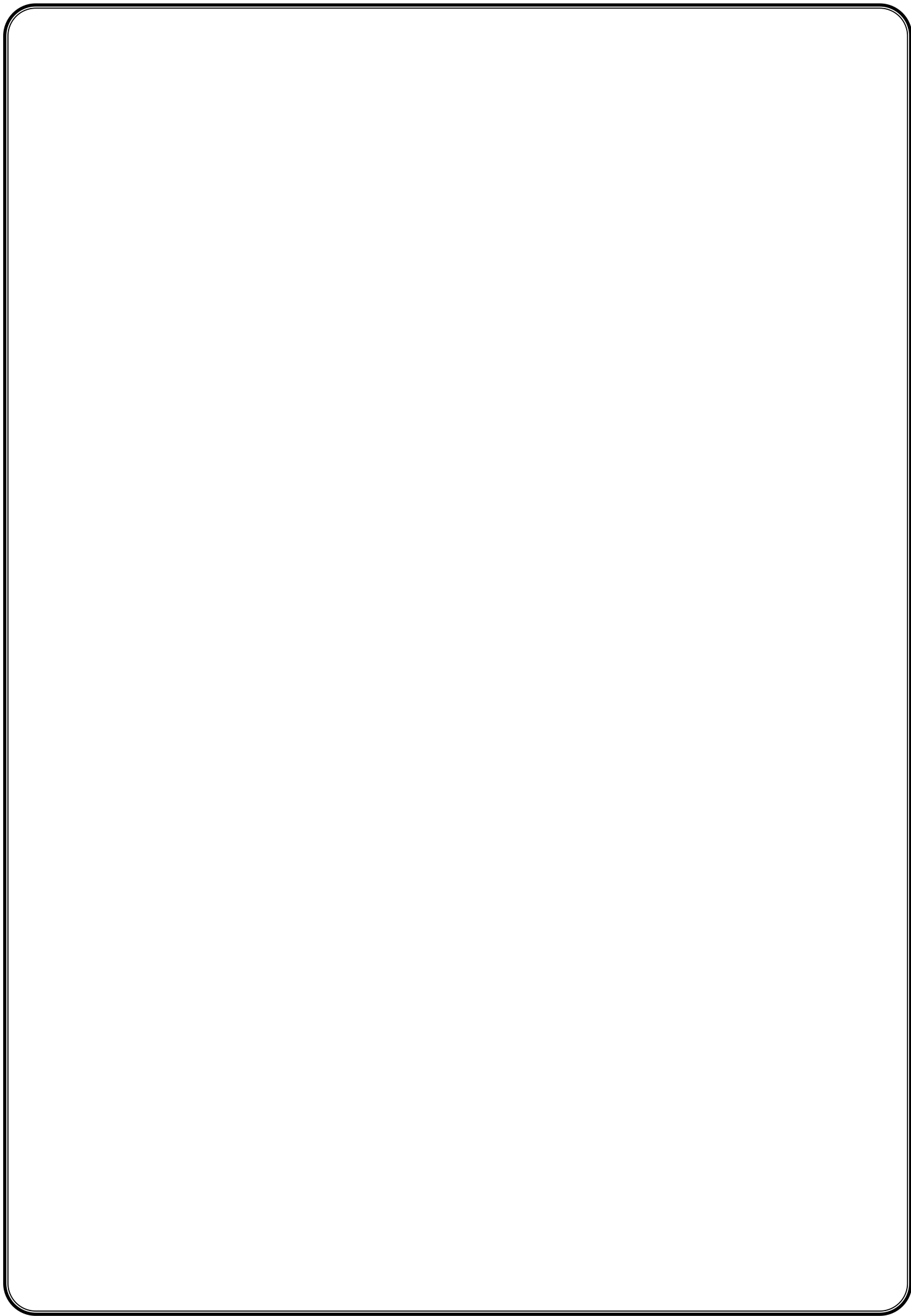


```
This is an <em>Example</em> of using the emphasis tag<br>
<font face="Times New Roman">Example</font> <br>
<font face="Times New Roman" size="4" color="#ff0000">Example</font>
<hr width="25%" color="#6699ff" size="6"/>
<i>Example</i>

<menu>
<li type="disc">List item 1</li>
<li type="circle">List item 2</li>
<li type="square">List item 3</li>
</menu>
<ol type="i">
<li>List item 1</li>
<li>List item 2</li>
<li>List item 3</li>
<li>List item 4</li>
</ol>
</body>
</html>
```

Result :-

Program executed and output obtained successfully.



Program No.2

Link different HTML pages

Aim :- Create a HTML file to link to different HTML page which contains images, tables, and also link within a page.

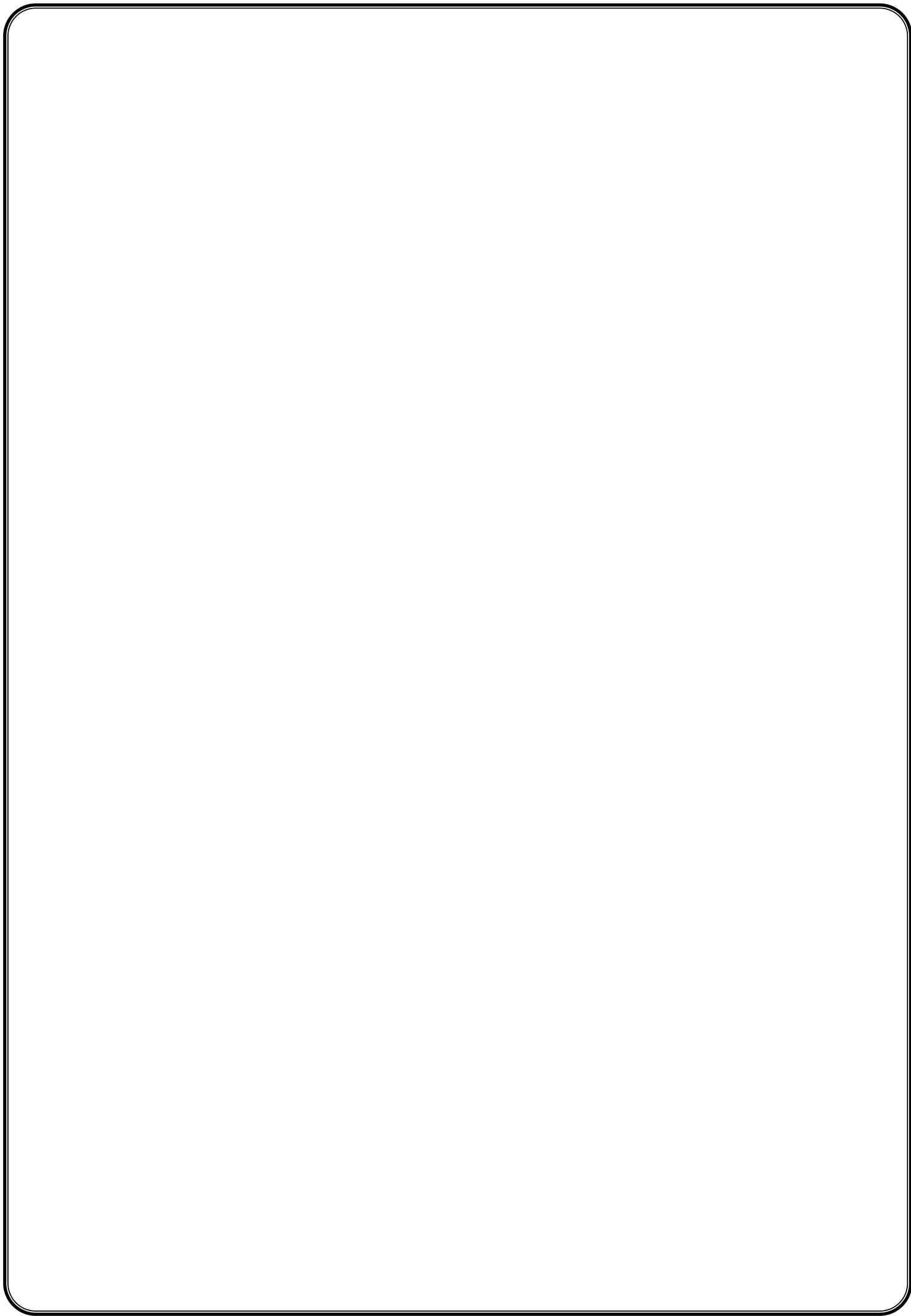
Algorithm :-

1. Open a text editor or HTML editor.
2. Start with an HTML5 doctype declaration: <!DOCTYPE html>.
3. Add the opening and closing html tags.
4. Add the head section within the html tags.
5. Within the head section, add a title tag to give the page a title.
6. Add a link tag within the head section to link to a stylesheet if needed.
7. Close the head section.
8. Add the body section within the html tags.
9. Add a header section within the body tags to include any header information you want.
10. Add a nav section within the body tags to create a navigation menu for the links.
11. Within the nav section, create a list of links using the ul and li tags.
12. Use the a tag within each li tag to link to the appropriate HTML pages.
13. Add a section tag within the body tags to contain the main content of the page.
14. Within the section tag, create any additional div or article tags needed to organize the content.
15. Add an img tag within the section tag to include any images needed.
16. Use the table tag to create any tables needed within the section tag.
17. Within the section tag, use the a tag to create links within the same page using href="#anchor" and id="anchor" to specify the location of the link.
18. Add a footer section within the body tags to include any footer information you want.
19. Close the body section and the html section.

Program :-

Home page

```
<!DOCTYPE html>
<html>
  <head>
    <title>Home</title>
  </head>
  <body>
    <center>
      <br>
      <table
border="1"cellspacing="10"cellpadding="20"width="100"height="100"bgcolor="powderblue
"bordercolor="yellow">
        <tr>
          <th>Item</th>
          <th>Link</th>
        </tr>
        <tr>
          <td>Biodata</td>
```



```
 <a href="file:///home/student/ page1">Click here</a></td> </tr> </table>  </center> </body> </html> |
```

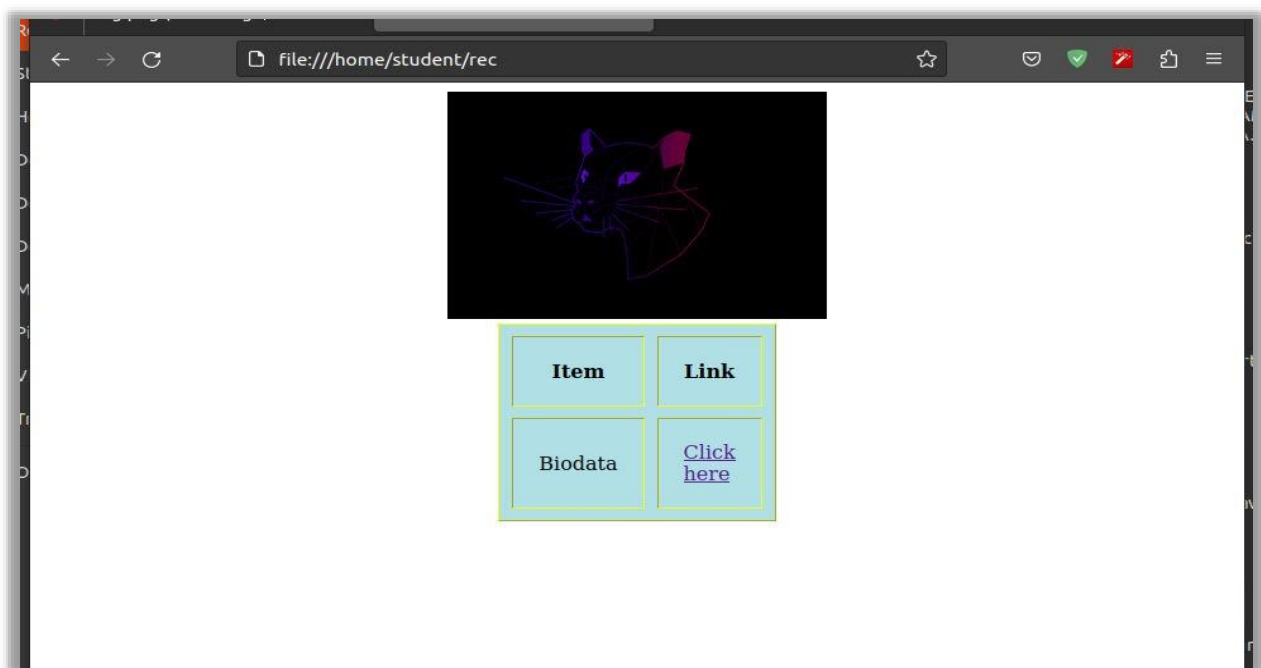
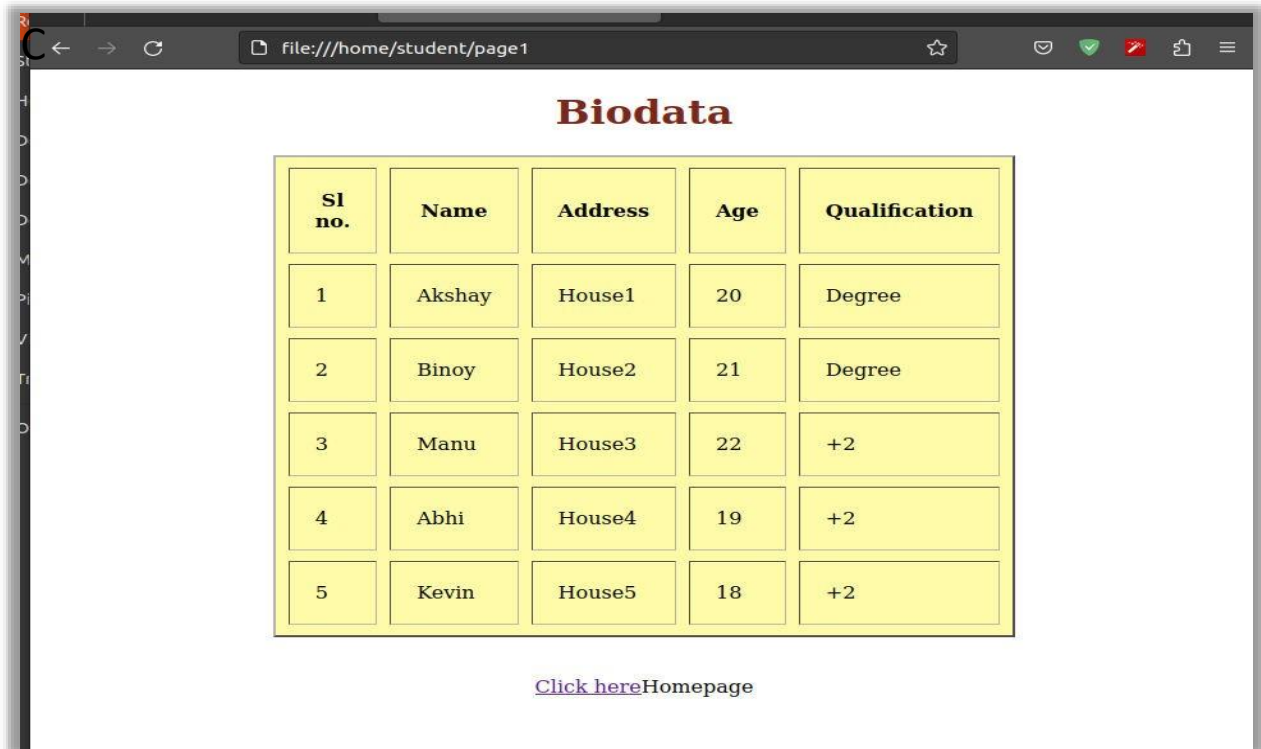
Linked page

```

<!DOCTYPE html>
<html>
  <head>
    <title>Page1</title>
  </head>
  <body>
    <center>
      <h1><font color="#7a271d">Biodata</font></h1>
      <table border="2"bgcolor="#fffba8"
cellspacing="10"cellpadding="20"width="200"height="300">
        <tr>
          <th>Sl no.</th>
          <th>Name</th>
          <th>Address</th>
          <th>Age</th>
          <th>Qualification</th>
        </tr>
        <tr>
          <td>1</td>
          <td>Akshay</td>
          <td>House1</td>
          <td>20</td>
          <td>Degree</td>
        </tr>
        <tr>
          <td>2</td>
          <td>Binoy</td>
          <td>House2</td>
          <td>21</td>
          <td>Degree</td>
        </tr>
        <tr>
          <td>3</td>
          <td>Manu</td>
          <td>House3</td>
          <td>22</td>
          <td>+2</td>
        </tr>
        <tr>
          <td>4</td>
          <td>Abhi</td>

```

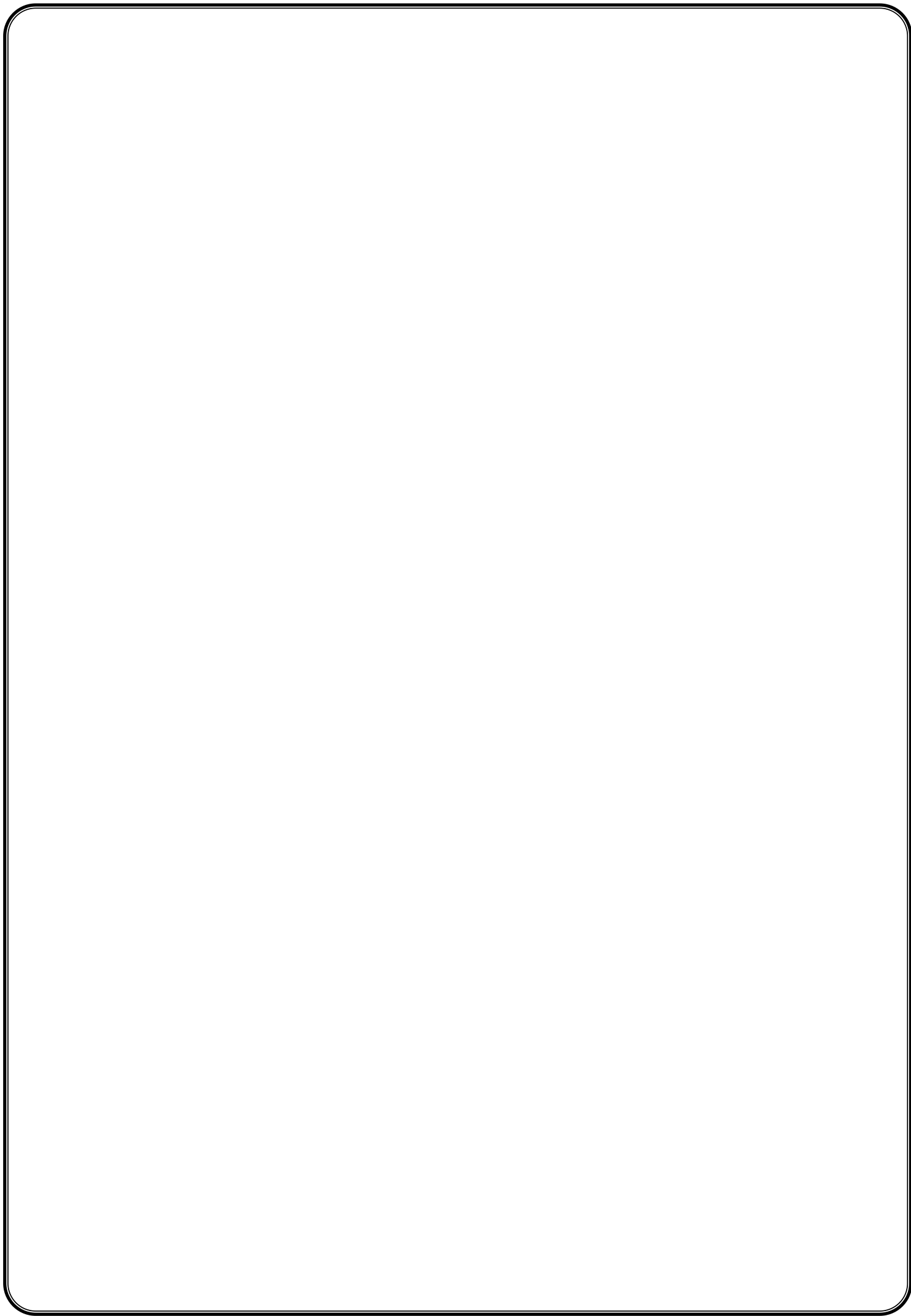
Output :-



```
        <td>House4</td>
        <td>19</td>
        <td>+2</td>
    </tr>
    <tr>
        <td>5</td>
        <td>Kevin</td>
        <td>House5</td>
        <td>18</td>
        <td>+2</td>
    </tr>
</table><br><br>
<a href="file:///home/student/rec">Click here</a>Homepage
</center>
</body>
</html>
```

Result :-

Program executed and output obtained successfully.



Program No.3

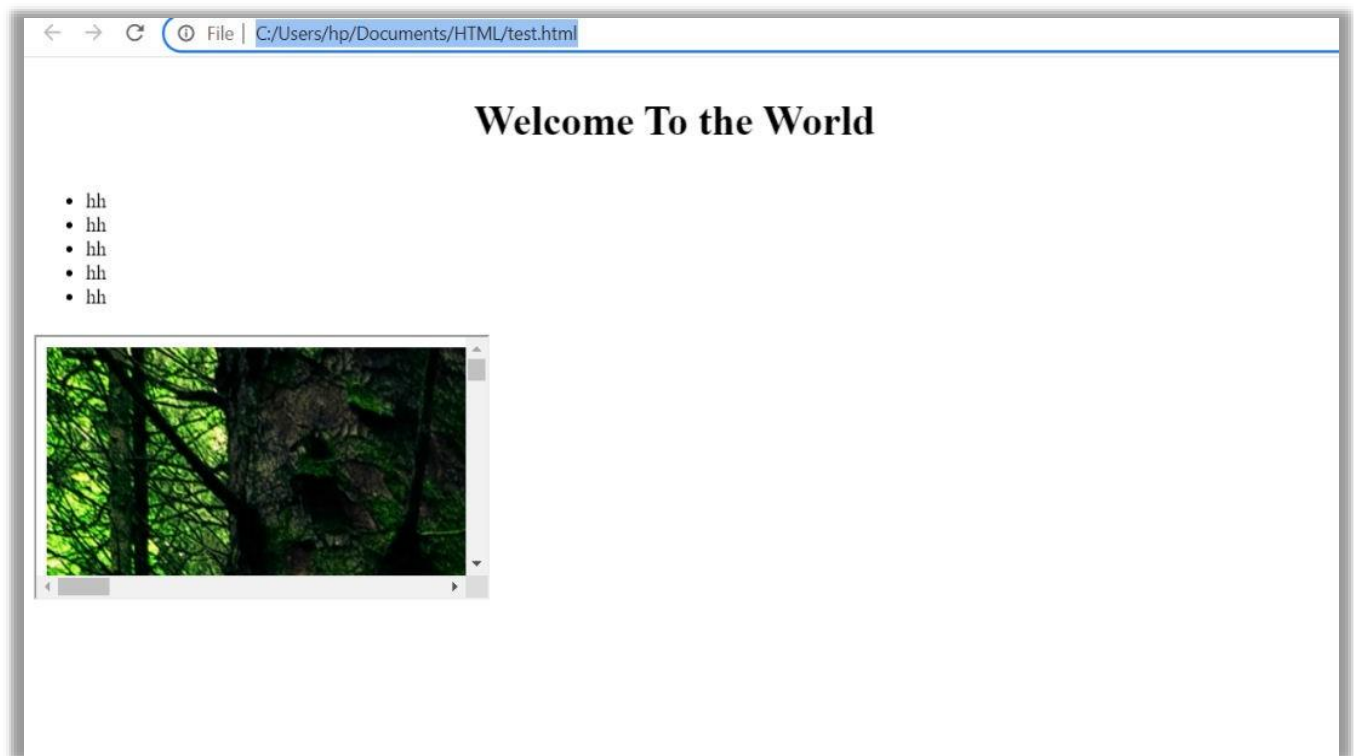
HTML page with different frames

Aim :- To create a HTML page with different types of frames such as floating frame,navigation frame & mixed frame.

Algorithm :-

- 1.Open a text editor or HTML editor.
- 2.Start with an HTML4 transitional doctype declaration: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">.
3. Add the opening and closing html tags.
4. Add the head section within the html tags.
5. Within the head section, add a title tag to give the page a title.
6. Close the head section.
7. Add the frameset section within the html tags.
- 8.Within the frameset section, specify the number and size of the frames using the frame and frameset tags.
9. Use the frame tag to define each frame and its contents.
- 10.Use the src attribute within the frame tag to specify the URL of the content to be displayed in the frame.
11. Use the name attribute within the frame tag to specify a name for each frame.
- 12.Use the cols or rows attribute within the frameset tag to specify the layout of the frames either horizontally or vertically.
- 13.Use the frameborder attribute within the frameset tag to specify whether or not the frames should have borders.
- 14.Use the border attribute within the frame tag to specify whether or not the frame content should have a border.
- 15.Use the scrolling attribute within the frame tag to specify whether or not the frame conten should have scrollbars.
- 16.Add a noframes section within the frameset section to display content for browsers that do not support frames.
- 17.Within the noframes section, add HTML code to display content for browsers that do not support frames.

Output :-



Program :-

```
<html lang="en">
<body>
<header>
<marquee id="head_marquee" direction="right" behavior="alternate" scrollamount="8"
onmouseover="this.stop()" onmouseout="this.start()">
<h1>Welcome To the World</h1>
</marquee>
</header>
<marquee direction="up" scrollamount="3" onmouseover="this.stop()" onmouseout="this.start()">
<ul>
<li>hh </li>
<li>hh </li>
<li>hh </li>
<li>hh </li>
<li>hh </li>
</ul>
</marquee>
<IFRAME NAME="float1" SRC="C:\Users\hp\Documents\HTML\float.html" WIDTH=350 HEIGHT=200
ALIGN=LEFT>
</IFRAME>
</body>
</html>
```

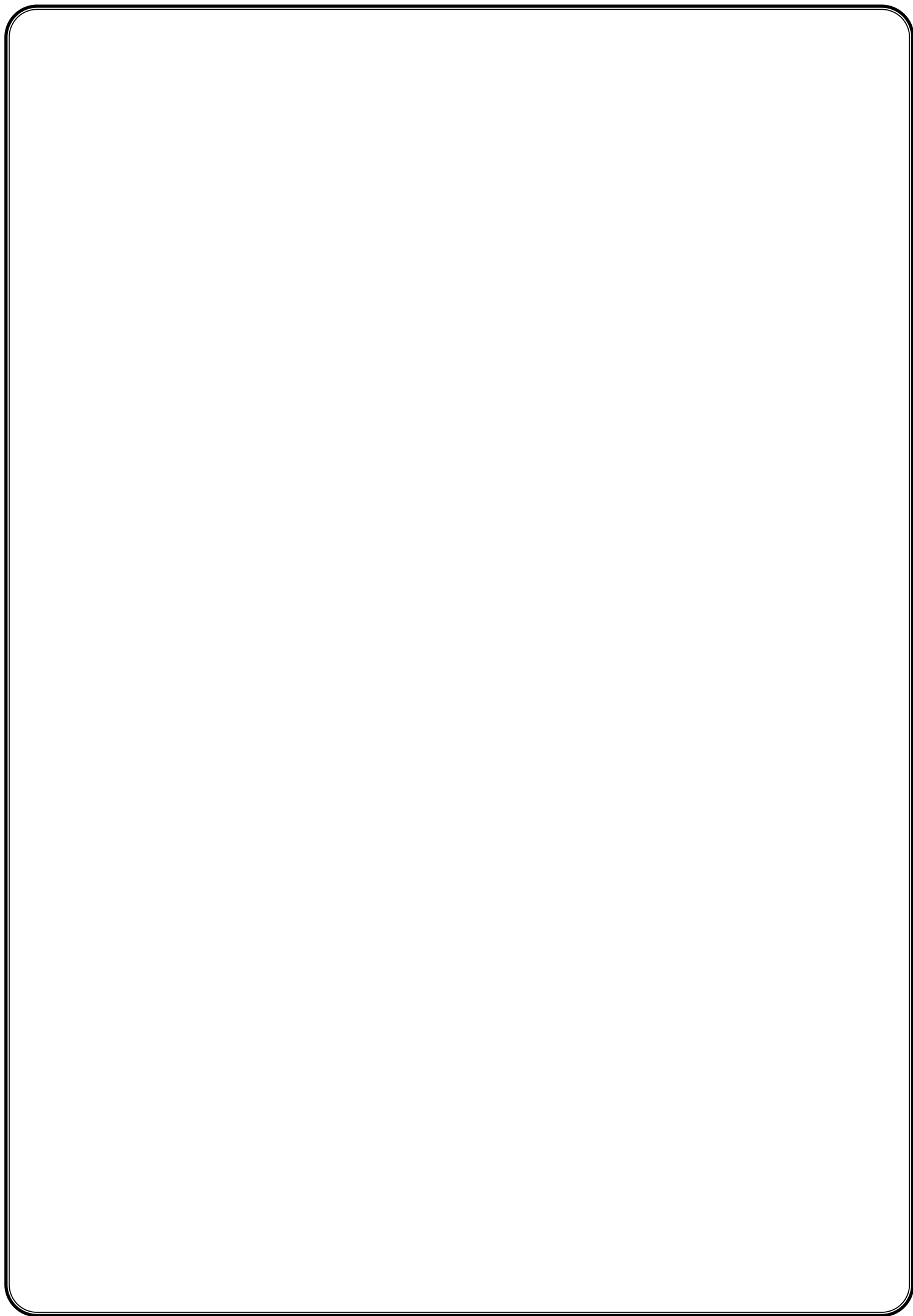
Iframe

```
<html>
<body>

</body>
</html>
```

Result :-

Program executed and output obtained successfully.



Program No.4

HTML files using different styles

Aim :- Create a HTML file by applying the different styles using inline, external & internal styleSheets

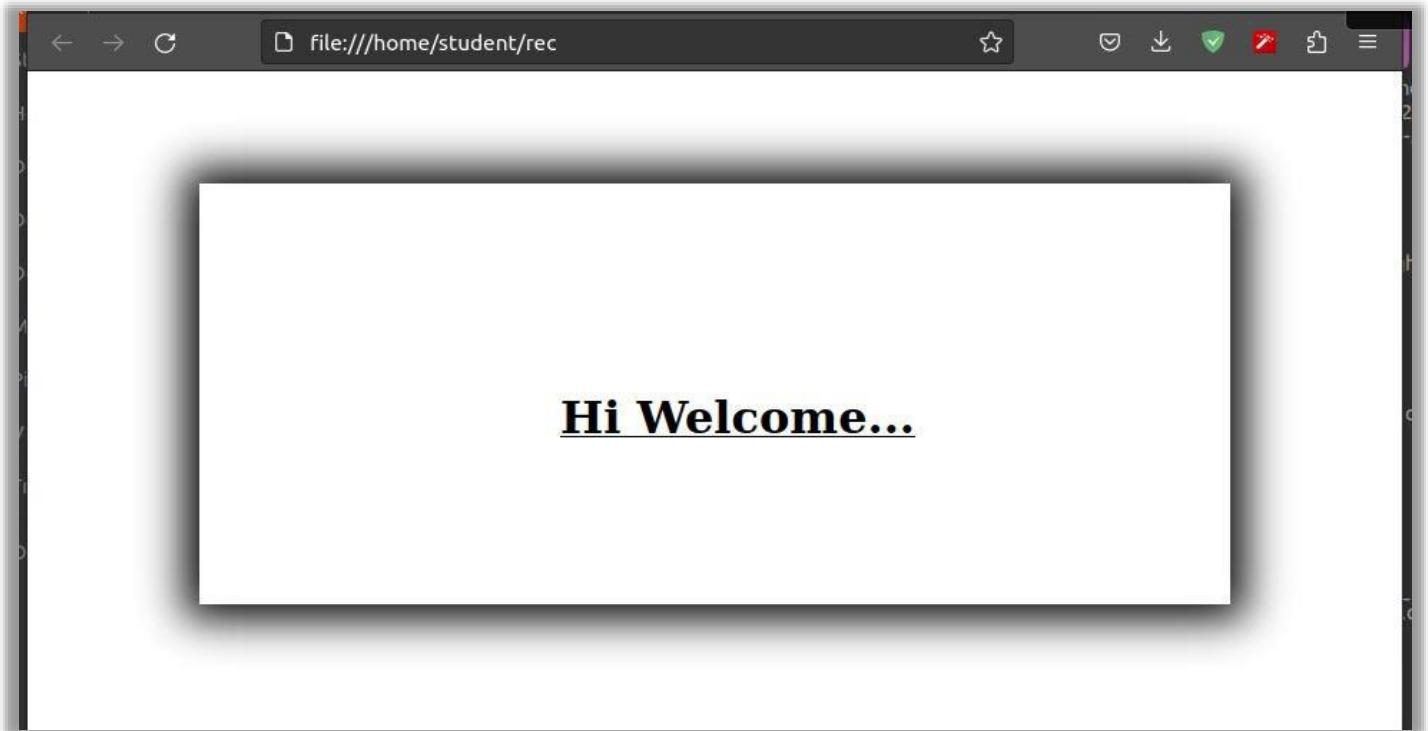
Algorithm :-

1. Open a text editor or HTML editor.
2. Start with an HTML5 doctype declaration: <!DOCTYPE html>.
3. Add the opening and closing html tags.
4. Add the head section within the html tags.
5. Within the head section, add a title tag to give the page a title.
6. Add a link tag to include an external style sheet in the head section. Use the rel attribute with a value of stylesheet, the type attribute with a value of text/css, and the href attribute with the file path or URL of the external style sheet. For example: <link rel="stylesheet" type="text/css" href="style.css">.
7. Add an style tag to include internal style sheet in the head section. Use the type attribute with a value of text/css. Within the style tag, write CSS rules to apply styles to HTML elements on the page.

Program :-

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>css1</title>
    <link rel="stylesheet" href="box.css">
    <style>
      body{height:100%;width:100%;margin:0;padding:0;background-
image:url(file:///home/student/pavi/web/bg1.jpg);background-position:center;background-
attachment:fixed;background-repeat:no-repeat;background-size:cover;}
      td{padding:1em;padding-left:3em;text-align:justify;}
      table{margin-left:auto;margin-right:auto;text-align:left;}
      h1{padding-top:1em;text-align:center;}
    </style>
  </head>
  <body>
    <div>
      <table class="box">
        <tbody>
          <tr>
            <td colspan="3"><h1><u>Hi
Welcome...</u></h1></td>
          </tr>
        </tbody>
      </table>
    </div>
  </body>
</html>
```

Output :-

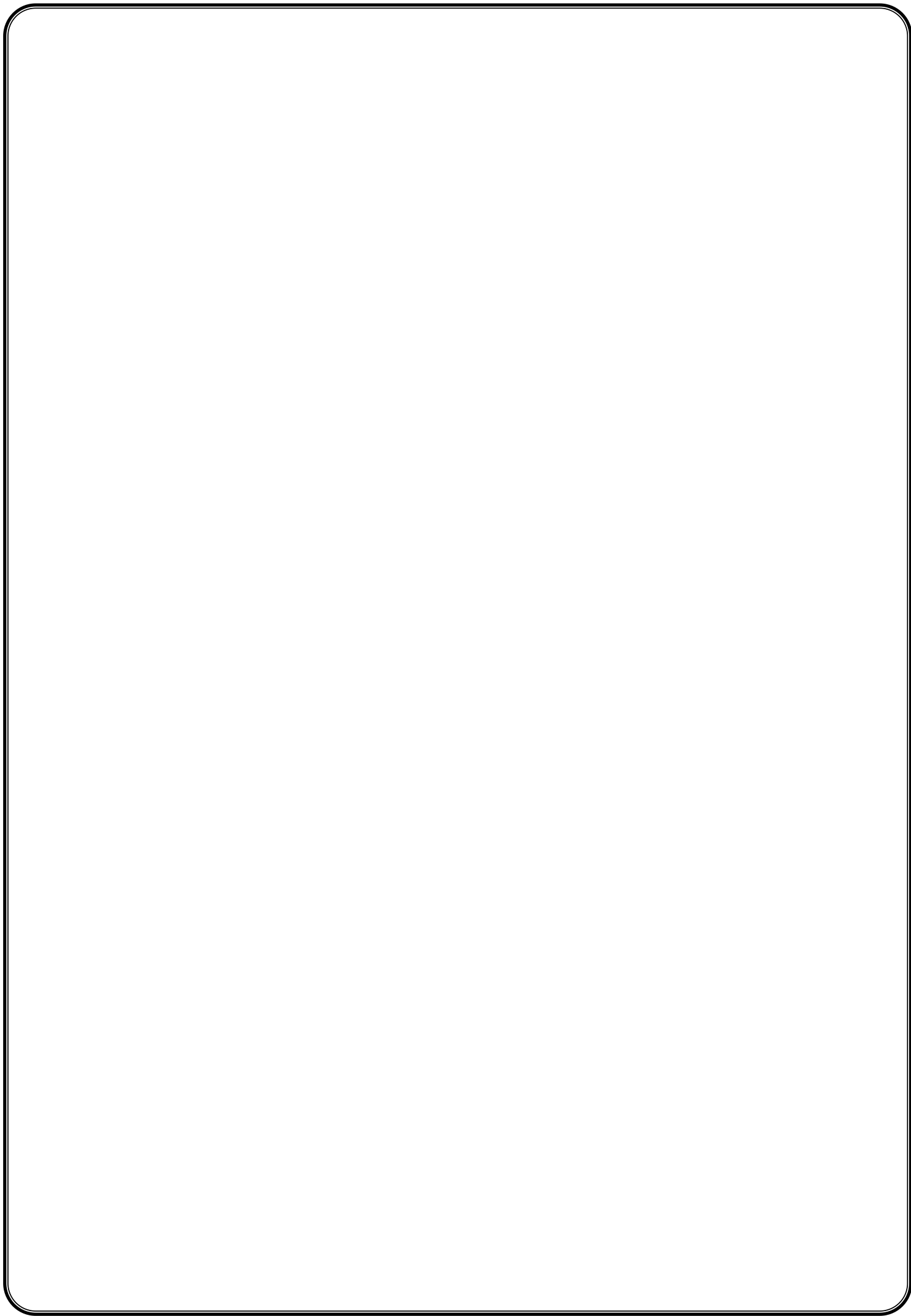


box.css

```
.box{width:75%;height:300px;background-color:white;box-shadow:0px 0px 30px 10px #000;margin-left:auto;margin-right:auto;margin-top:5em;border-collapse:collapse;}
```

Result :

Program executed and output obtained successfully.



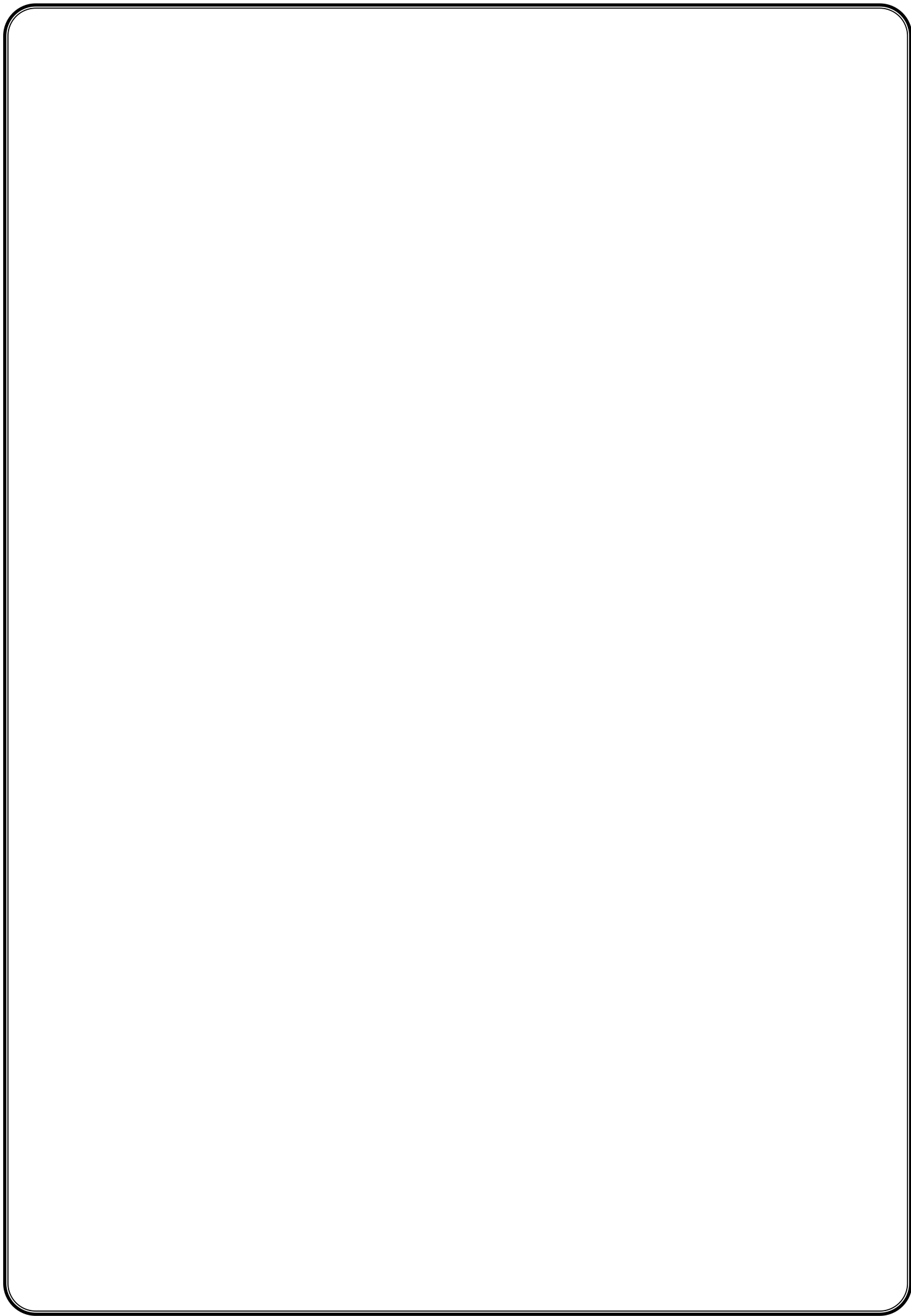
Program No.5

Registration form using HTML

Aim :- To create a registration form using HTML

Algorithm :-

1. Open a text editor or HTML editor.
2. Start with an HTML5 doctype declaration: `<!DOCTYPE html>`.
3. Add the opening and closing html tags.
4. Add the head section within the html tags.
5. Within the head section, add a title tag to give the page a title.
6. Close the head section.
7. Add the body section within the html tags.
8. Use the form tag to define the registration form.
9. Use the action attribute within the form tag to specify the URL of the page that will handle the form data.
10. Use the method attribute within the form tag to specify the HTTP method to use for submitting the form data (e.g. "get" or "post").
11. Use the input tag to define each input field within the form.
12. Use the type attribute within the input tag to specify the type of input field (e.g. "text", "password", "email", "checkbox", etc.).
13. Use the name attribute within the input tag to specify a name for each input field.
14. Use the placeholder attribute within the input tag to specify a placeholder text for each input field.
15. Use the required attribute within the input tag to specify whether or not each input field is required.
16. Use the label tag to create a label for each input field.
17. Use the for attribute within the label tag to associate the label with its corresponding input field using the id attribute.
18. Use the button tag to create a submit button for the form.
19. Use the type attribute within the button tag to specify the type of button (e.g. "submit").
20. Add any additional HTML elements or styling as desired.



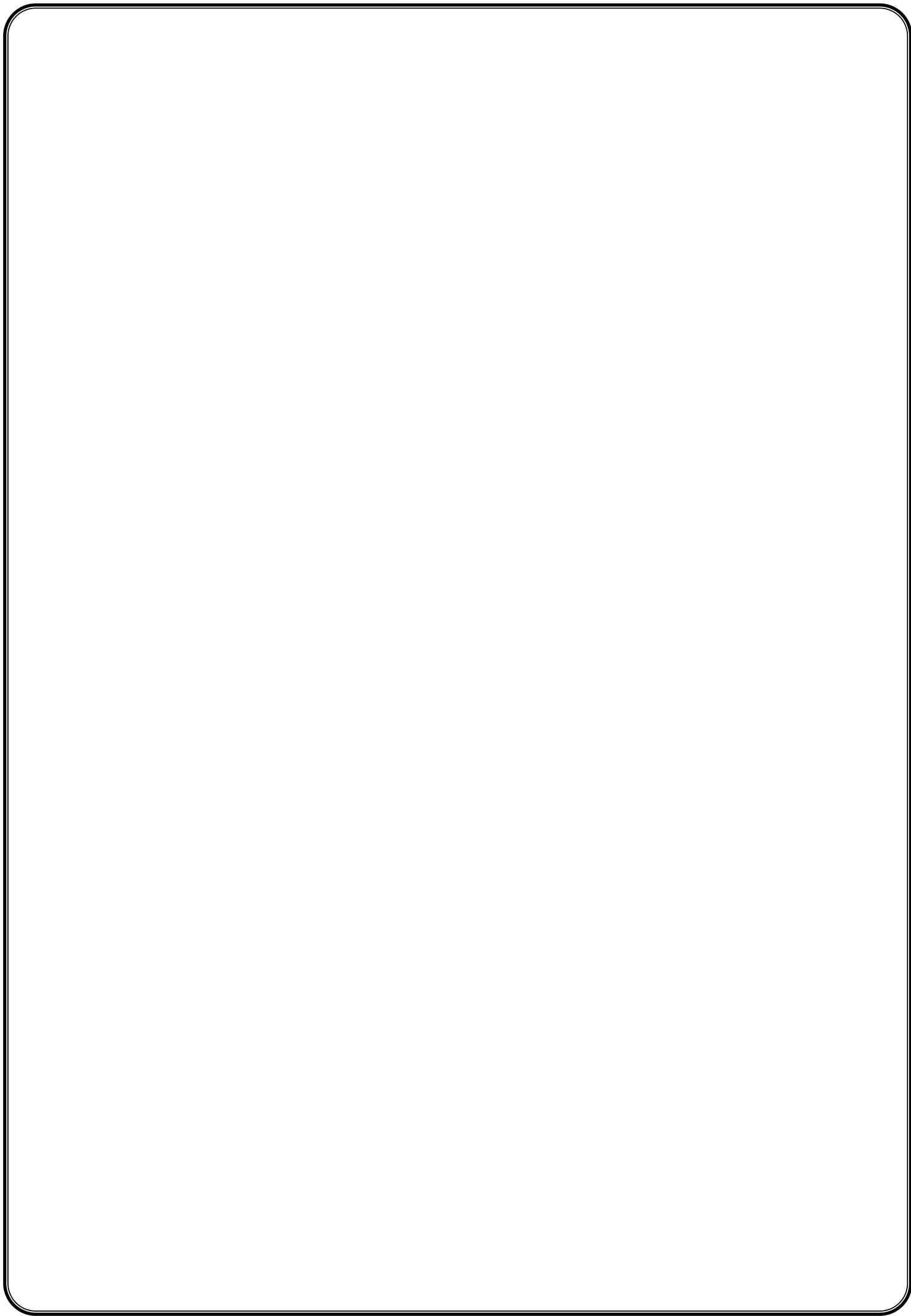
Program :-

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>freshersform</title>
    <style>
      h4{background-color:yellow;color:red;width:fit-content}
      table{background-color:rgb(212, 209, 209)}
      .reset{color:white;background-color: red;height:2em;width:5em;border:0ch}
      .submit{color:white;background-color:green;height:2em;width:5em;border:0ch;}
    </style>
  </head>
  <body>
    <header>
      <marquee id="headm"
direction="left"behavior=""scrollamount="5"onmouseover="this.stop()"onmouseout="this.st
art()" >

        <h4><b>WELCOME TO FRESHERS WORLD!!!</b></h4>
      </marquee>
    </header>
    <center>
      <table border="1"cellpadding="10"cellspacing="0"width="75%"height="">
        <tr>
          <th colspan="3"><b>REGISTRATION FORM</b></th>
        </tr>
        <tr>
          <td>NAME</td>
          <td colspan="2"><input type="text" id="name" name="name"/></td>
        </tr>
        <tr>
          <td>AGE</td>
          <td colspan="2">
            <select>
              <option>SELECT</option>
              <option></option>
            </select>
          </td>
        </tr>
        <tr>
          <td>PASSWORD</td>
          <td colspan="2"><input type="password" id="pass" name="pass"/></td>
        </tr>
        <tr>
          <td>RETYPE PASSWORD</td>
          <td colspan="2"><input type="password" id="pass" name="pass"/></td>
        </tr>
        <tr>
          <td rowspan="4">SELECT YOUR SECURITY QUESTION</td>
          <td colspan="2"><input type="radio" id="ques" name="ques"/>What is your pet
name?</td>
        </tr>
        <tr>
          <td colspan="2"><input type="radio" id="ques" name="ques"/>Who is your best

```



```

friend?</td>
</tr>
<tr>
<td colspan="2"><input type="radio" id="ques" name="ques"/>What is your
favourite color?</td>
</tr>
<td colspan="2"><input type="radio" id="ques" name="ques"/>Who is your
favourite teacher?</td>
</tr>
<tr>
<td>ANSWER OF SECURITY QUESTION</td>
<td colspan="2"><input type="text" id="que" name="que"/></td>
</tr>
<tr>
<td>EMAIL ID</td>
<td colspan="2"><input type="email" id="email" name="email"/></td>
</tr>
<tr>
<td>LANGUAGES KNOWN</td>
<td colspan="2">
<input type="checkbox" id="lan" name="lan"/>MALAYALAM
<input type="checkbox" id="lan" name="lan"/>ENGLISH
<input type="checkbox" id="lan" name="lan"/>HINDI
</td>
</tr>
<tr>
<td rowspan="2">PHONE NUMBER</td>
<td>Home</td>
<td>Office</td>
<tr>
<td><input type="tel" id="home" name="home"/></td>
<td><input type="tel" id="office" name="office"/></td>
</tr>
</tr>
<tr>
<td>UPLOAD CV</td>
<td colspan="2"><input type="file" id="cv" name="cv" value="Choose File"/></td>
</tr>
<tr>
<td><center><input
type="reset" id="reset" name="reset" value="Reset" class="reset"/></center></td>
<td colspan="2"><center><input
type="submit" id="submit" name="submit" value="Submit" class="submit"/></center></td>
</tr>
</table>
</center>
</body>
</html>

```

Output :-

← → ↻

file:///home/student/rec

☆

🔒

⬇

✅

🔴

📄

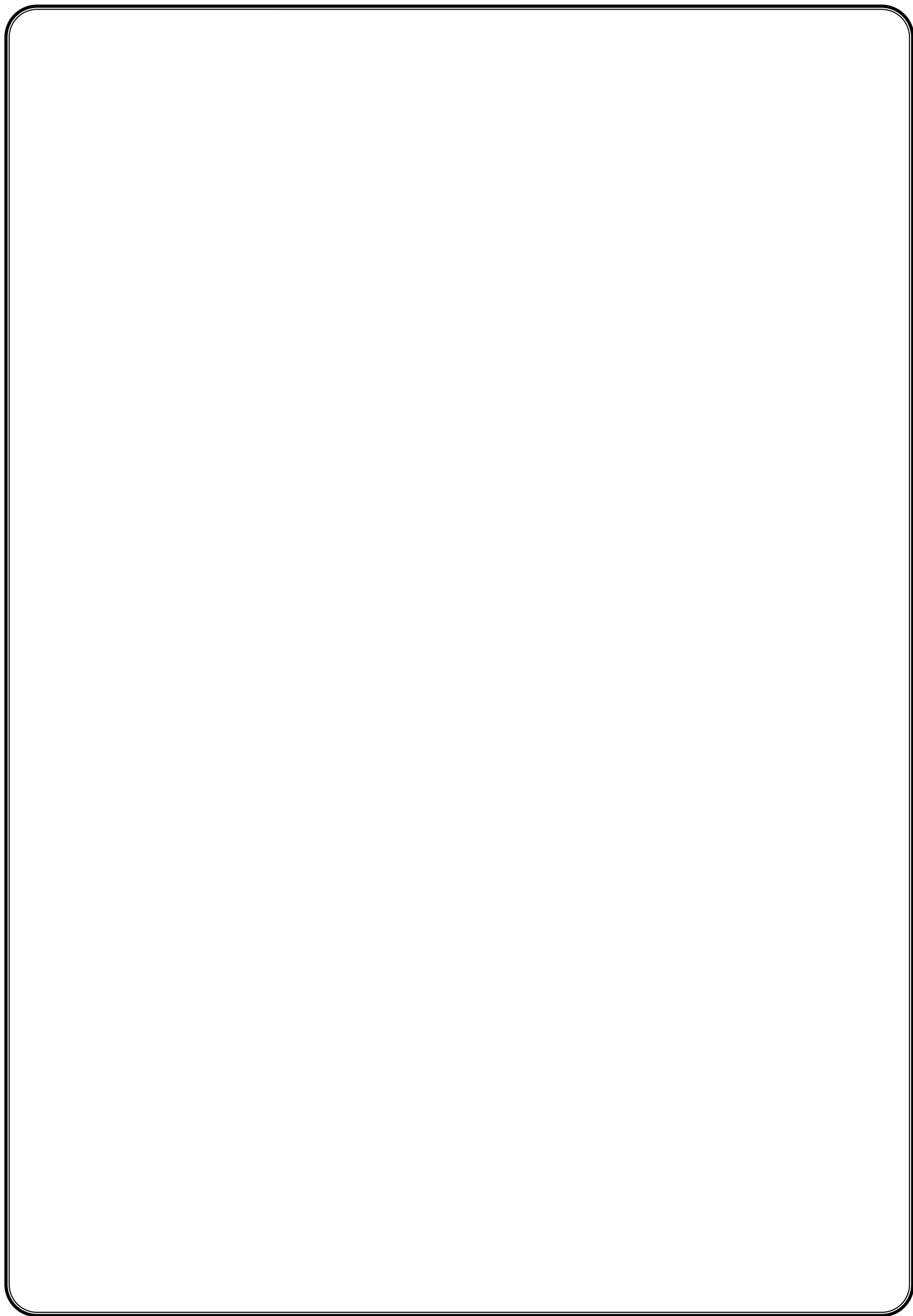
☰

WELCOME TO FRESHERS WORLD!!!

REGISTRATION FORM	
NAME	<input type="text"/>
AGE	<div>SELECT ▾</div>
PASSWORD	<input type="password"/>
RETYPE PASSWORD	<input type="password"/>
SELECT YOUR SECURITY QUESTION	<div><input type="radio"/> What is your pet name?</div>
	<div><input type="radio"/> Who is your best friend?</div>
	<div><input type="radio"/> What is your favourite color?</div>
	<div><input type="radio"/> Who is your favourite teacher?</div>
ANSWER OF SECURITY QUESTION	<input type="text"/>
EMAIL ID	<input type="text"/>
LANGUAGES KNOWN	<div><input type="checkbox"/> MALAYALAM <input type="checkbox"/> ENGLISH <input type="checkbox"/> HINDI</div>
PHONE NUMBER	<div>Home</div>
	<div>Office</div>
UPLOAD CV	<div><div>Browse...</div>No file selected.</div>
<div>Reset</div>	<div>Submit</div>

Result :-

Program executed and output obtained successfully.



Program No.6

Predefined functions in a string and math object in javascript

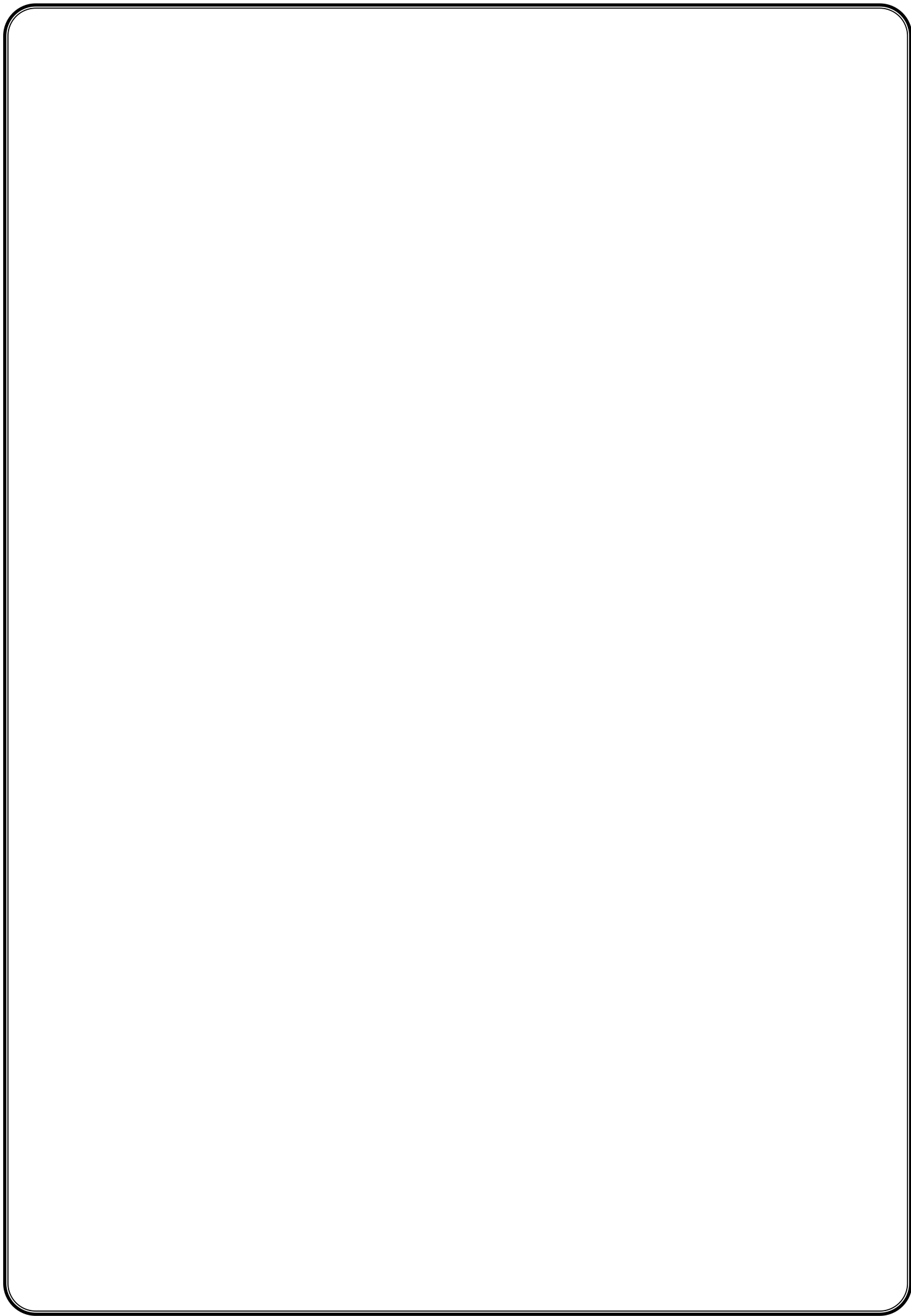
Aim:- To create a HTML page to explain the use of various predefined functions in a string and math object in javascript.

Algorithm :-

- 1.Create a new HTML file.Program executed and output obtained successfully.
2. Add a title to the page, such as "JavaScript String and Math Functions".
- 3.Create a new section in the page to explain the String functions. Add a heading to this section.
- 4.List the various String functions that you want to explain. For example, you could include functions such as charAt(), concat(), indexOf(), and replace()
- 5.For each function, provide a brief description of what it does and how it is used. You could also include an example of how to use the function in JavaScript code.
- 6.Create a new section in the page to explain the Math functions. Add a heading to this section.
- 7.List the various Math functions that you want to explain. For example, you could include functions such as abs(), sqrt(), ceil(), and floor().
- 8.For each function, provide a brief description of what it does and how it is used. You could also include an example of how to use the function in JavaScript code.
- 9.Save and preview the HTML file in a web browser to ensure that everything is working correctly.
- 10.Optionally, you can add styling and formatting to the page using CSS to make it more visually appealing.

Program :-

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>String and Math functions</title>
<style>
div {
text-align: center;
border-bottom: 2px solid black;
padding-bottom: 2em;
}
h2 {
text-transform: uppercase;
text-align: center;
margin-top: 2em;
width: 100%;
padding: 5px;
color: #222324;
font-family: 'Courier New', Courier, monospace;
box-shadow: 0 4px 10px 0 rgba(0, 0, 0, 0.2), 0 4px 20px 0 rgba(0, 0, 0, 0.19);
}
table,
tr,
td,
```

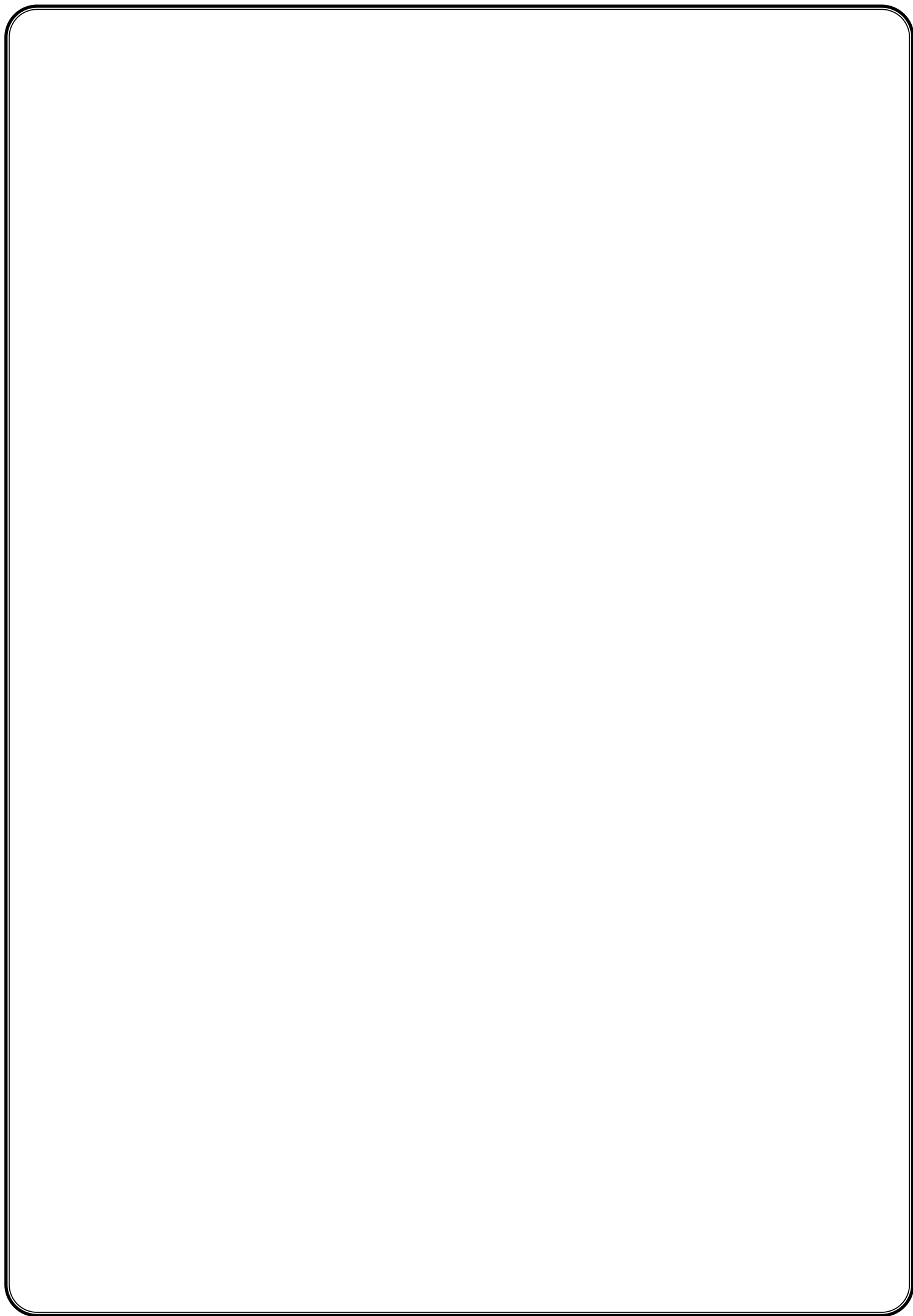



```

,th {
padding: 1em;
}
table {
width: 100%;
}
th {
text-align: left;
}
.selector {

padding: .5em;
background-color: #002607;
color: white;
border: none;
outline: none;
border-radius: 5px;
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans
Unicode', Geneva, Verdana, sans-serif;
}
li {
text-align: left;
margin-top: 5px;
padding: 10px;
}
ul {
background-color: #1a1b1c;
color: white;
}
</style>
</head>
<body>
<div>
<h2>String Functions</h2>
<table>
<tbody>
<tr>
<th>Enter a text</th>
<td> <input type="text" id="input_srting" name="input_s"></td>
</tr>
<tr>
<th>Search a text</th>
<td> <input type="text" id="input_search" name="input_search"></td>
</tr>
<tr>
<th>Slice start</th>
<td><input type="text" id="slice_s" name="slice_s"></td>
</tr>
<tr>
<th>Slice end</th>
<td><input type="text" id="slice_e" name="slice_e"></td>
</tr>
<tr>
<th>Replace the word</th>

```

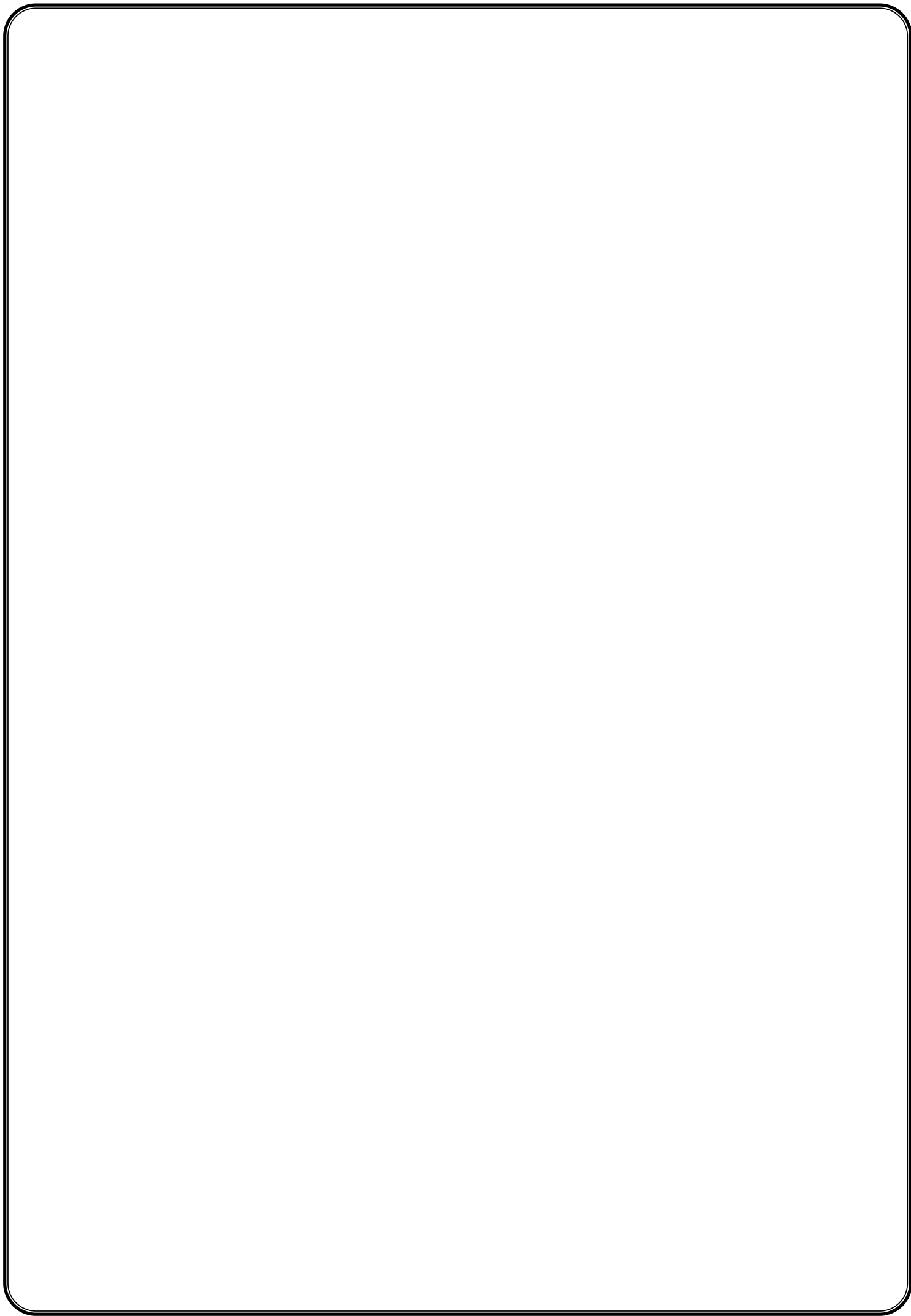


```

<td><input type="text" id="replace_s"></td>
</tr>
</tbody>
</table>
<input type="button" value="Generate" class="selector" onclick="string_func()">
<br><br>
<ul id="output"> </ul>
</div>
<div>
<h2>Math Functions<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>String and Math functions</title>
<style>
div {

text-align: center;
border-bottom: 2px solid black;
padding-bottom: 2em;
}
h2 {
text-transform: uppercase;
text-align: center;
margin-top: 2em;
width: 100%;
padding: 5px;
color: #222324;
font-family: 'Courier New', Courier, monospace;
box-shadow: 0 4px 10px 0 rgba(0, 0, 0, 0.2), 0 4px 20px 0 rgba(0, 0, 0, 0.19);
}
table,
tr,
td,
th {
padding: 1em;
}
table {
width: 100%;
}
th {
text-align: left;
}
.selector {
padding: .5em;
background-color: #002607;
color: white;
border: none;
outline: none;
border-radius: 5px;
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans
Unicode', Geneva, Verdana, sans-serif;
}

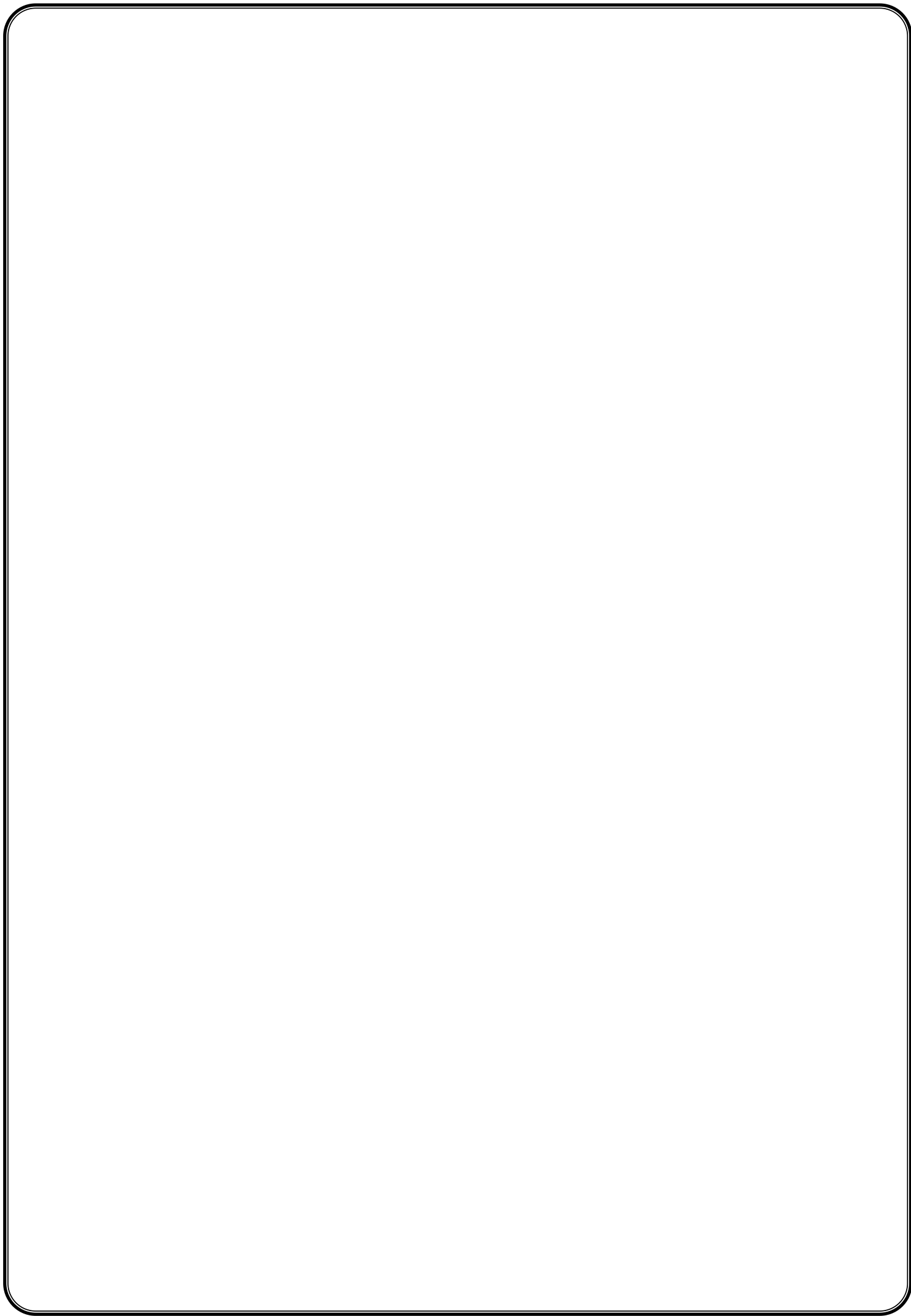
```



```

li {
text-align: left;
margin-top: 5px;
padding: 10px;
}
ul {
background-color: #1a1b1c;
color: white;
}
</style>
</head>
<body>
<div>
<h2>String Functions</h2>
<table>
<tbody>
<tr>
<th>Enter a text</th>
<td> <input type="text" id="input_srting" name="input_s"></td>
</tr>
<tr>
<th>Search a text</th>
<td> <input type="text" id="input_search" name="input_search"></td>
</tr>
<tr>
<th>Slice start</th>
<td><input type="text" id="slice_s" name="slice_s"></td>
</tr>
<tr>
<th>Slice end</th>
<td><input type="text" id="slice_e" name="slice_e"></td>
</tr>
<tr>
<th>Replace the word</th>
<td><input type="text" id="replace_f"></td>
</tr>
<tr>
<th>Replace with</th>
<td><input type="text" id="replace_s"></td>
</tr>
</tbody>
</table>
<input type="button" value="Generate" class="selector" onclick="string_func()">
<br><br>
<ul id="output"> </ul>
</div>
<div>
<h2>Math Functions</h2>
<table>
<tbody>
<tr>

```

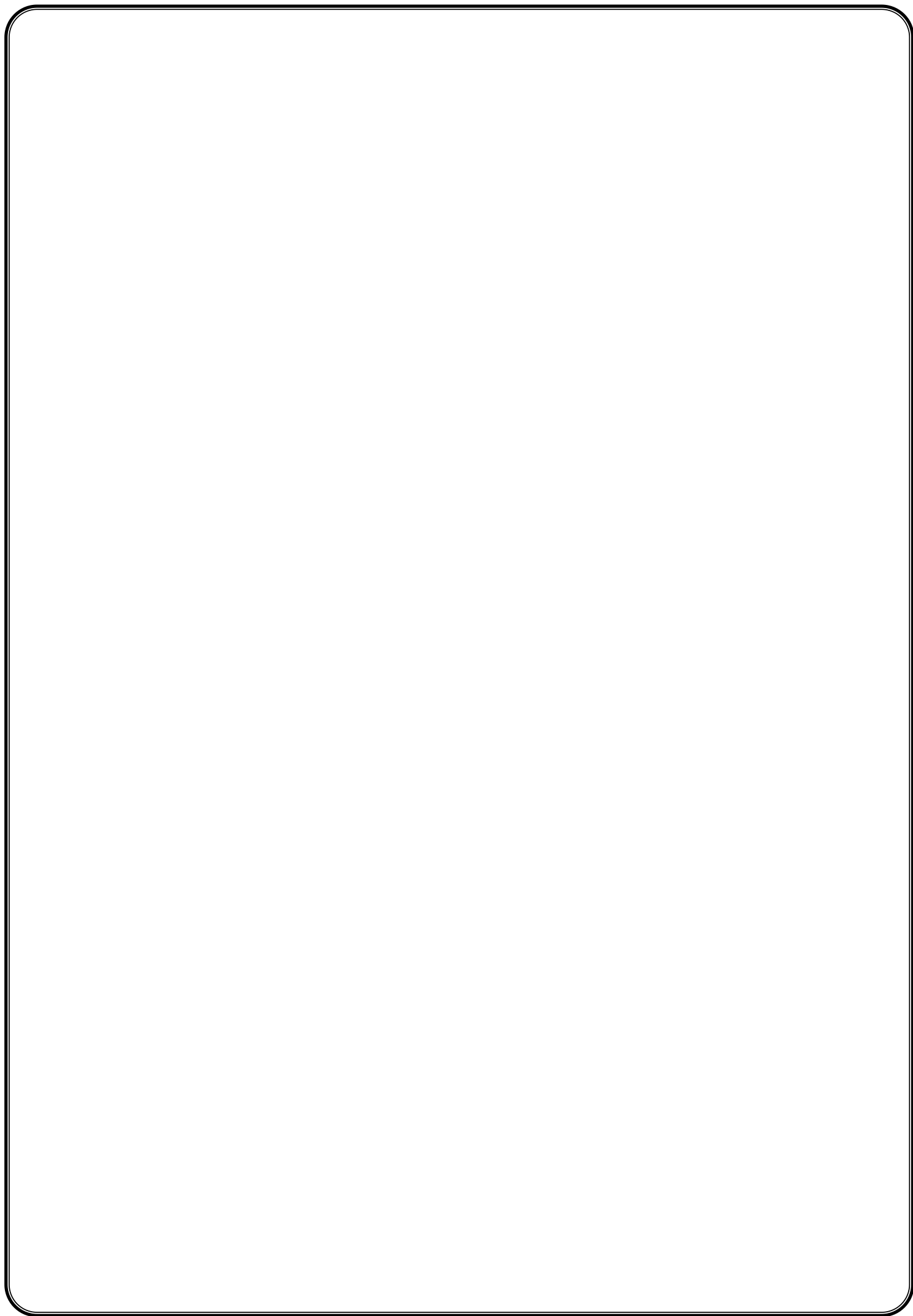


```

<th>Enter a Number</th>
<td> <input type="text" id="input_num"></td>
</tr>
<tr>
<th>Power term</th>
<td> <input type="text" id="input_pow"></td>
</tr>
</tbody>
</table>
<input type="button" value="Generate" class="selector" onclick="math_func()">
<br><br>
<ul id="output_math"> </ul>
</div>
<script>
function string_func() {
let input = document.getElementById('input_srtng').value;
let search = document.getElementById('input_search').value;
let slice_s = document.getElementById('slice_s').value;
let slice_e = document.getElementById('slice_e').value;
let replace_f = document.getElementById('replace_f').value;
let replace_s = document.getElementById('replace_s').value;
let li_O = '<li>';
let li_C = '</li>';
let list_items;
list_items = li_O + 'Length of string is : ' + input.length + li_C;
list_items += li_O + 'First index of ' + search + ' is ' + input.indexOf(search) + li_C;
list_items += li_O + 'Last index of ' + search + ' is ' + input.lastIndexOf(search) + li_C;
list_items += li_O + 'Index with search starting position 5 of ' + search + ' is ' +
input.indexOf(search, 5) + li_C;
list_items += li_O + 'Search ' + search + ' is ' + input.search(search) + li_C;
list_items += li_O + 'Slice[' + slice_s + ' : ] is ' + input.slice(slice_s) + li_C;
list_items += li_O + 'Substring[' + slice_s + ' : ' + slice_e + '] is ' + input.substring(slice_s,
slice_e) + li_C;
list_items += li_O + 'Substr[' + slice_s + ' : ] is ' + input.substr(slice_s) + li_C;
list_items += li_O + 'Substr[' + slice_s + ' : ] with length ' + slice_e + ' is ' +
input.substr(slice_s, slice_e) + li_C;

list_items += li_O + 'replace ' + replace_f + ' with ' + replace_s + ' is ' +
input.replace(replace_f, replace_s) + li_C;
list_items += li_O + 'Uppercase : ' + input.toUpperCase() + li_C;
list_items += li_O + 'lowercase : ' + input.toLowerCase() + li_C;
document.getElementById('output').innerHTML = (list_items);
}
function math_func() {
let input = document.getElementById('input_num').value;
let power = document.getElementById('input_pow').value;
let li_O = '<li>';
let li_C = '</li>';
let list_items;
list_items = li_O + 'Value of PI : ' + Math.PI + li_C;
list_items += li_O + 'Round of '+input+' is : ' + Math.round(input) + li_C;
list_items += li_O + '+input+' raise to '+power+' is : ' + Math.pow(input,power) + li_C;
list_items += li_O + 'Square root of '+input+' is : ' + Math.sqrt(input) + li_C;

```




```

list_items += li_O + 'Absolute value of '+input+' is : ' + Math.abs(input) + li_C;
list_items += li_O + 'Ceil value of '+input+' is : ' + Math.ceil(input) + li_C;
list_items += li_O + 'Floor value of '+input+' is : ' + Math.floor(input) + li_C;
list_items += li_O + 'Sin value of '+input+' is : ' + Math.sin(input* Math.PI / 180) + li_C;
list_items += li_O + 'Cos value of '+input+' is : ' + Math.cos(input* Math.PI / 180) + li_C;
list_items += li_O + 'Minimum of (0, 150, 30, 20, -8, -200) is : ' + Math.min(0, 150, 30,
20, -8, -200) + li_C;
list_items += li_O + 'Maximum of (0, 150, 30, 20, -8, -200) is : ' + Math.max(0, 150, 30,
20, -8, -200) + li_C;
document.getElementById('output_math').innerHTML = (list_items);
}
</script>
</body>
.+
</html></h2>
<table>
<tbody>
<tr>
<th>Enter a Number</th>
<td> <input type="text" id="input_num"></td>
</tr>
<tr>
<th>Power term</th>
<td> <input type="text" id="input_pow"></td>
</tr>
</tbody>
</table>
<input type="button" value="Generate" class="selector" onclick="math_func()">
<br><br>
<ul id="output_math"> </ul>
</div>
<script>
function string_func() {
let input = document.getElementById('input_srtng').value;
let search = document.getElementById('input_search').value;
let slice_s = document.getElementById('slice_s').value;
let slice_e = document.getElementById('slice_e').value;
let replace_f = document.getElementById('replace_f').value;
let replace_s = document.getElementById('replace_s').value;
let li_O = '<li>';
let li_C = '</li>';
let list_items;
list_items = li_O + 'Length of string is : ' + input.length + li_C;

list_items += li_O + 'First index of ' + search + ' is ' + input.indexOf(search) + li_C;
list_items += li_O + 'Last index of ' + search + ' is ' + input.lastIndexOf(search) + li_C;
list_items += li_O + 'Index with search starting position 5 of ' + search + ' is ' +
input.indexOf(search, 5) + li_C;
list_items += li_O + 'Search ' + search + ' is ' + input.search(search) + li_C;
list_items += li_O + 'Slice[' + slice_s + ' : ] is ' + input.slice(slice_s) + li_C;
list_items += li_O + 'Substring[' + slice_s + ' : ' + slice_e + '] is ' +

```

Output :-

← → ↺ 🏠

file:///home/student/ag/math.html

... 🛡️ ☆

⬇️ 📄 🗑️ ☰

STRING FUNCTIONS

Enter a text

how are you

Search a text

are

Slice start

5

Slice end

7

Replace the word

how

Replace with

how old

Generate

- Length of string is : 11
- First index of are is 4
- Last index of are is 4
- Index with search starting position 5 of are is -1
- Search are is 4
- Slice[5 :] is re you

← → ↺ 🏠

file:///home/student/ag/math.html

... 🛡️ ☆

⬇️ 📄 🗑️ ☰

- Slice[5 :] is re you
- Substring[5 : 7] is re
- Substr[5 :] is re you
- Substr[5 :] with length 7 is re you
- replace how with how old is how old are you
- Uppercase : HOW ARE YOU
- lowercase : how are you

MATH FUNCTIONS

Enter a Number

12

Power term

3

Generate

- Value of PI : 3.141592653589793
- Round of 12 is : 12
- 12 raised to 3 is : 1728

← → ↺ 🏠

file:///home/student/ag/math.html

... 🛡️ ☆

⬇️ 📄 🗑️ ☰

Enter a Number

12

Power term

3

Generate

- Value of PI : 3.141592653589793
- Round of 12 is : 12
- 12 raise to 3 is : 1728
- Square root of 12 is : 3.4641016151377544
- Absolute value of 12 is : 12
- Ceil value of 12 is : 12
- Floor value of 12 is : 12
- Sin value of 12 is : 0.20791169081775931
- Cos value of 12 is : 0.9781476007338057
- Minimum of (0, 150, 30, 20, -8, -200) is : -200
- Maximum of (0, 150, 30, 20, -8, -200) is : 150

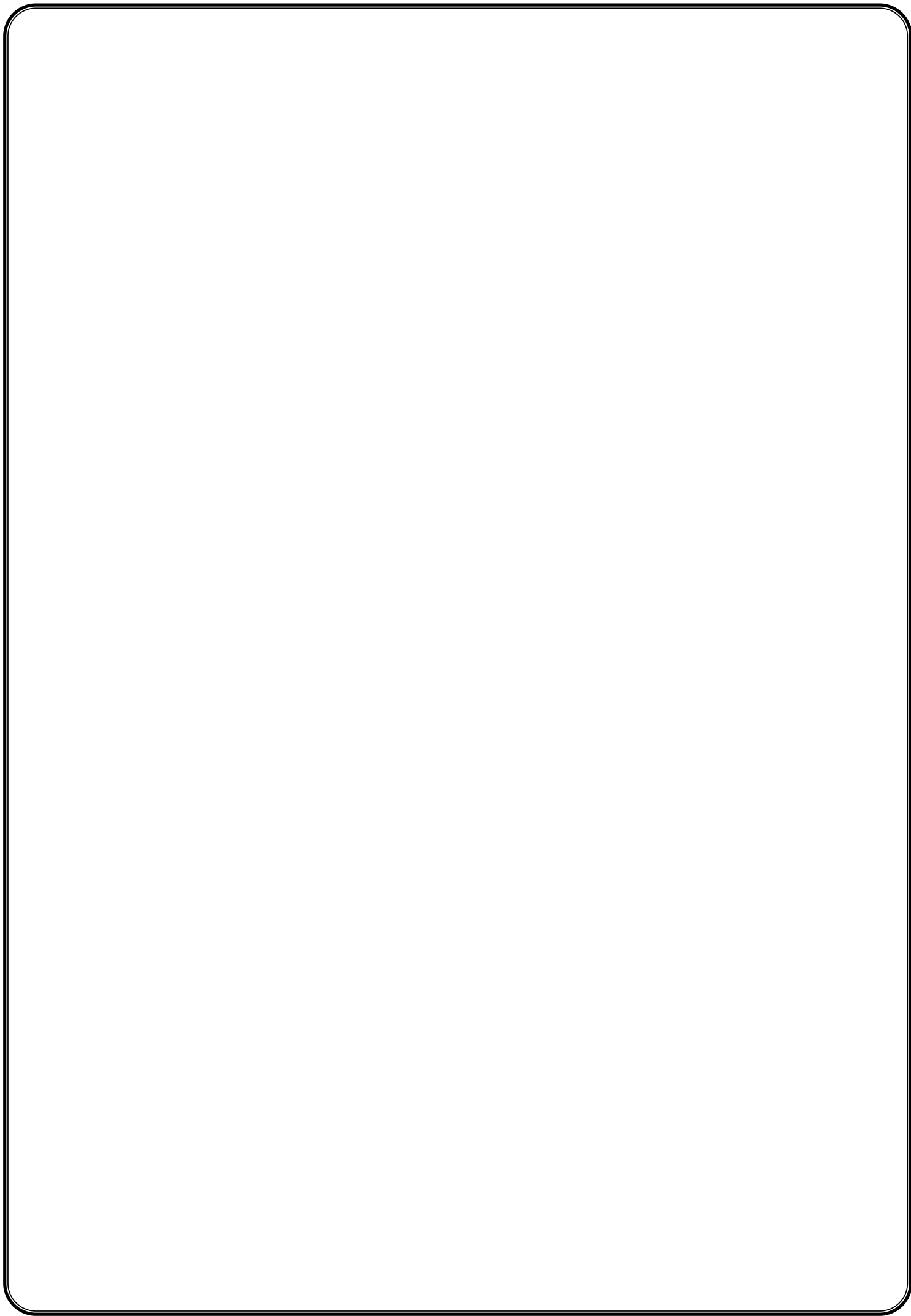
```

input.substring(slice_s,
slice_e) + li_C;
list_items += li_O + 'Substr[' + slice_s + ' :]' is ' + input.substr(slice_s) + li_C;
list_items += li_O + 'Substr[' + slice_s + ' :]' with length ' + slice_e + ' is ' +
input.substr(slice_s, slice_e) + li_C;
list_items += li_O + 'replace ' + replace_f + ' with ' + replace_s + ' is ' +
input.replace(replace_f, replace_s) + li_C;
list_items += li_O + 'Uppercase : ' + input.toUpperCase() + li_C;
list_items += li_O + 'lowercase : ' + input.toLowerCase() + li_C;
document.getElementById('output').innerHTML = (list_items);
}
function math_func() {
let input = document.getElementById('input_num').value;
let power = document.getElementById('input_pow').value;
let li_O = '<li>';
let li_C = '</li>';
let list_items;
list_items = li_O + 'Value of PI : ' + Math.PI + li_C;
list_items += li_O + 'Round of '+input+' is : ' + Math.round(input) + li_C;
list_items += li_O + '+input+' raise to '+power+' is : ' + Math.pow(input,power) + li_C;
list_items += li_O + 'Square root of '+input+' is : ' + Math.sqrt(input) + li_C;
list_items += li_O + 'Absolute value of '+input+' is : ' + Math.abs(input) + li_C;
` list_items += li_O + 'Ceil value of '+input+' is : ' + Math.ceil(input) + li_C;
list_items += li_O + 'Floor value of '+input+' is : ' + Math.floor(input) + li_C;
list_items += li_O + 'Sin value of '+input+' is : ' + Math.sin(input* Math.PI / 180) + li_C;
list_items += li_O + 'Cos value of '+input+' is : ' + Math.cos(input* Math.PI / 180) + li_C;
list_items += li_O + 'Minimum of (0, 150, 30, 20, -8, -200) is : ' + Math.min(0, 150, 30,
20, -8, -200) + li_C;
list_items += li_O + 'Maximum of (0, 150, 30, 20, -8, -200) is : ' + Math.max(0, 150, 30,
20, -8, -200) + li_C;
document.getElementById('output_math').innerHTML = (list_items);
}
</script>
</body>
. +
</html>

```

Result :-

Program executed and output obtained successfully.



Program No.7

Calender using Javascript

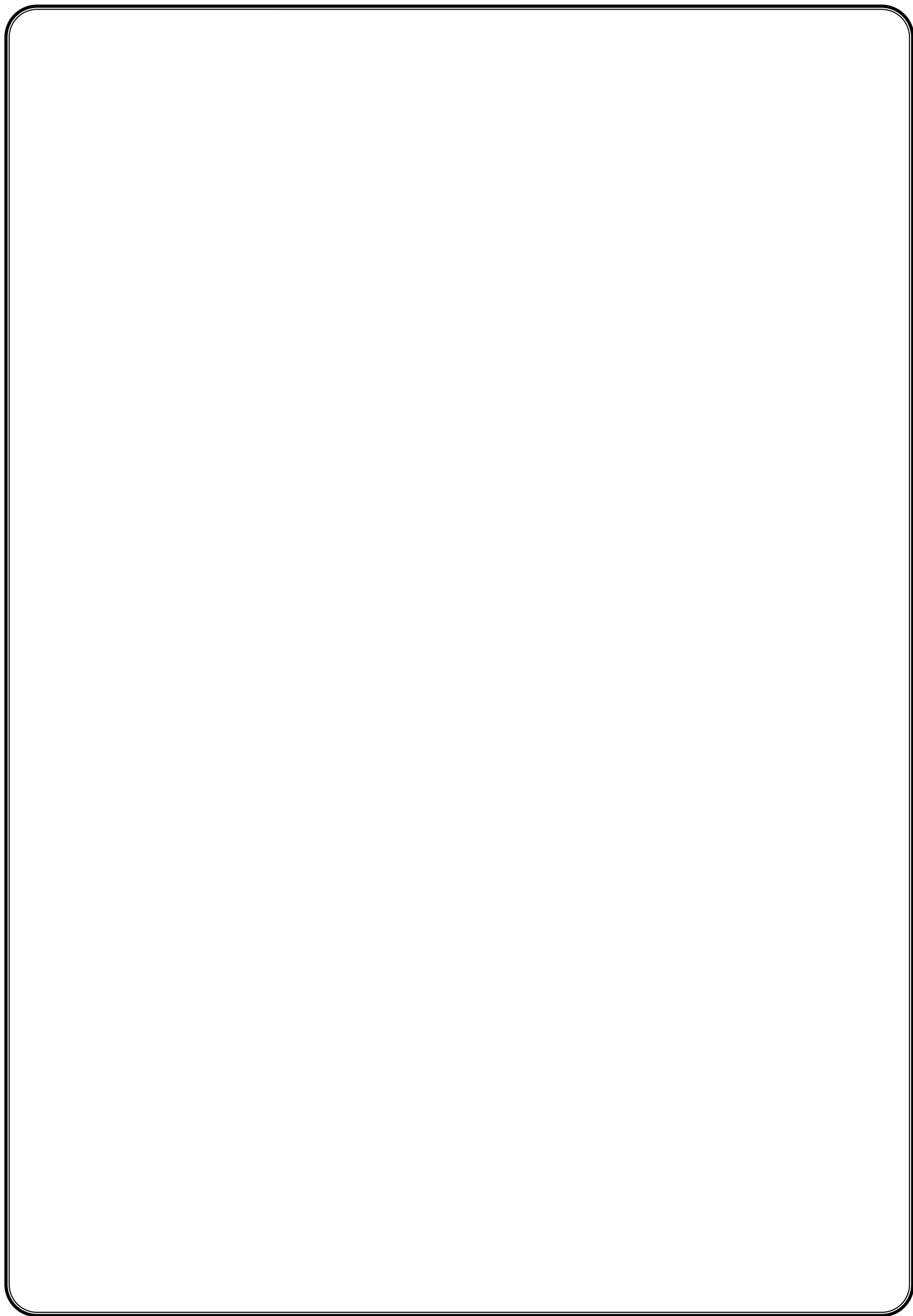
Aim :- To generate the calendar using JavaScript code by getting the year from the user
Program executed and output obtained successfully.

Algorithm :-

- 1.Create a new HTML file with a form that includes an input field for the user to enter the year.
- 2.Add a button to the form that triggers a JavaScript function when clicked.
- 3.In the JavaScript function, retrieve the year entered by the user from the input field.
- 4.Use the year to create a new Date object using the Date() constructor.
- 5.Use the getFullYear() method to retrieve the year from the Date object.
- 6.Use a loop to iterate through each month of the year. For each month, do the following:
- 7.Create a new Date object with the year and month.
- 8.Use the toLocaleDateString() method to format the date object as a string in the format "MMMM YYYY".
- 9.Create a table element with the days of the week as column headings and add it to the page.
- 10.Use the getDay() method to determine the day of the week for the first day of the month.
- 11.Use the getDate() method to determine the number of days in the month.
- 12.Use a loop to generate a row for each week of the month. For each week, do the following:
- 13.Use a loop to generate a cell for each day of the week. For each day, do the following:
- 14.If the cell is for a day before the first day of the month or after the last day of the month, leave it blank.
- 15.Otherwise, display the date in the cell.
- 16.Add styling to the calendar using CSS to make it more visually appealing.

Program :-

```
<html lang="en">
<head>
<meta charset="UTF-8">Program executed and output obtained successfully.
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Calender</title>
<style>
s.election {
text-align: center;
margin-bottom: 2em;
background-color:rgb(174, 179, 181);
width: 100%;
}
#year_get,#month_get{
padding:5px; width:75%;
}
input{ padding:
5px;
}
t.abselection {
```



```

margin-left: auto;
margin-right: auto;
width: 100%;
}
table{
padding: 5px;
}
tr{
}
.selrow{
border: none;
}
td,th{
padding: 10px;

border: 1px solid black;
}
.selcol{
padding:0px;
border: none;
text-align: center;
}
#calender{
border-collapse: collapse;
margin-left: auto;
margin-right: auto;
}
#resetbtn{
background-color:red;
border: none;
color: white;
}
#subtn{
background-color:green;
border: none;
color: white;
}
</style>
</head>
<body>
<div class="selection">
<table class="tabselection" cellspacing="10">
<tbody>
<tr class="selrow">
<td class="selcol"><input id="year_get" type="number" placeholder="year">

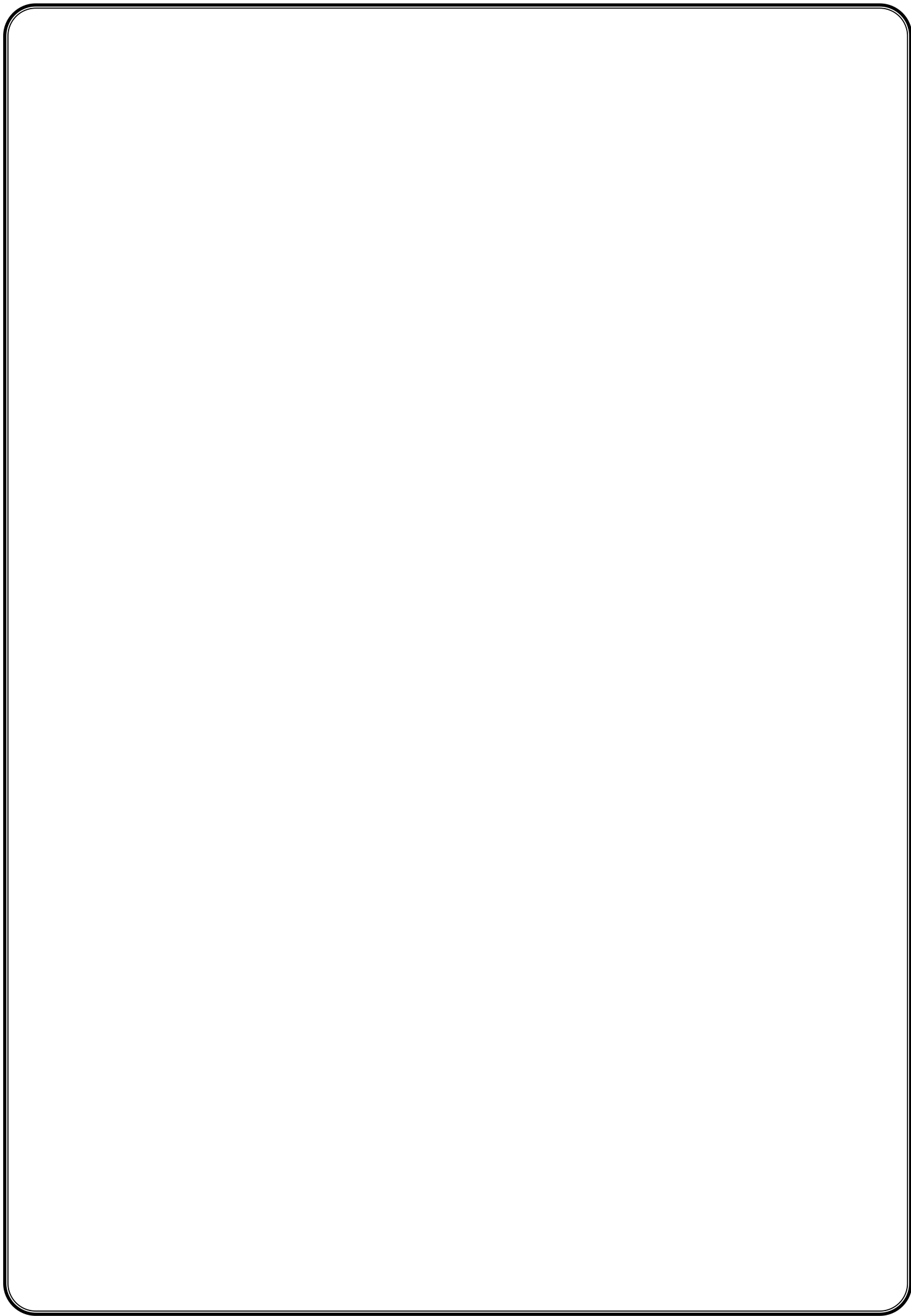
</td>

<td class="selcol"><input id="month_get" type="number"

placeholder="month"> </td>

</tr>

```




```

<tr class="selrow">
<td colspan="2" class="selcol"><input onclick="generate()" type="button"

id="subtn" value="Generate Calender"> </td>

</tr>
<tr class="selrow">
<td colspan="2" class="selcol"><input type="reset" value="reset"

id="resetbtn" onclick="location.reload();"> </td>

</tr>
</tbody>
</table>
</div>
<div id="content"> </div>
</body>
<script>
function generate() {

var init_content = "<table BORDER=1 id='calender'><tr><th>SUN</th>

<th>MON</th><th>TUE</th><th>WED</th><th>THU</th><th>FRI</th><th>SAT</th></tr>
<tr>"

var year_get = document.getElementById("year_get").value; var
month_get = document.getElementById("month_get").value;
month_get -=1;
var date = new Date(year_get,month_get);
var day = date.getDay();
for (var i = 0; i < day; i++) {
init_content += "<td></td>";
}
while (date.getMonth() == month_get) {
init_content += "<td>" + date.getDate() + "</td>";
if (date.getDay() % 7 == 6) {
init_content += "</tr><tr>";
}
date.setDate(date.getDate() + 1);
}
while(date.getDay() % 7 != 6 && date.getDay() % 7 != 0){
init_content += "<td></td>";
date.setDate(date.getDate() + 1);
}
if(date.getDay() % 7 > 0){
init_content += "<td></td></tr>";
}
init_content += "</table>"
document.getElementById("content").innerHTML = init_content;
}
</script>
</html>

```

Output :-

←

→

🔄

📄 File | /home/student/MCA/cal2.html

➤

☆

🖨

👤

Update ⋮

📧 Gmail

📺 YouTube

📍 Maps

2023

2

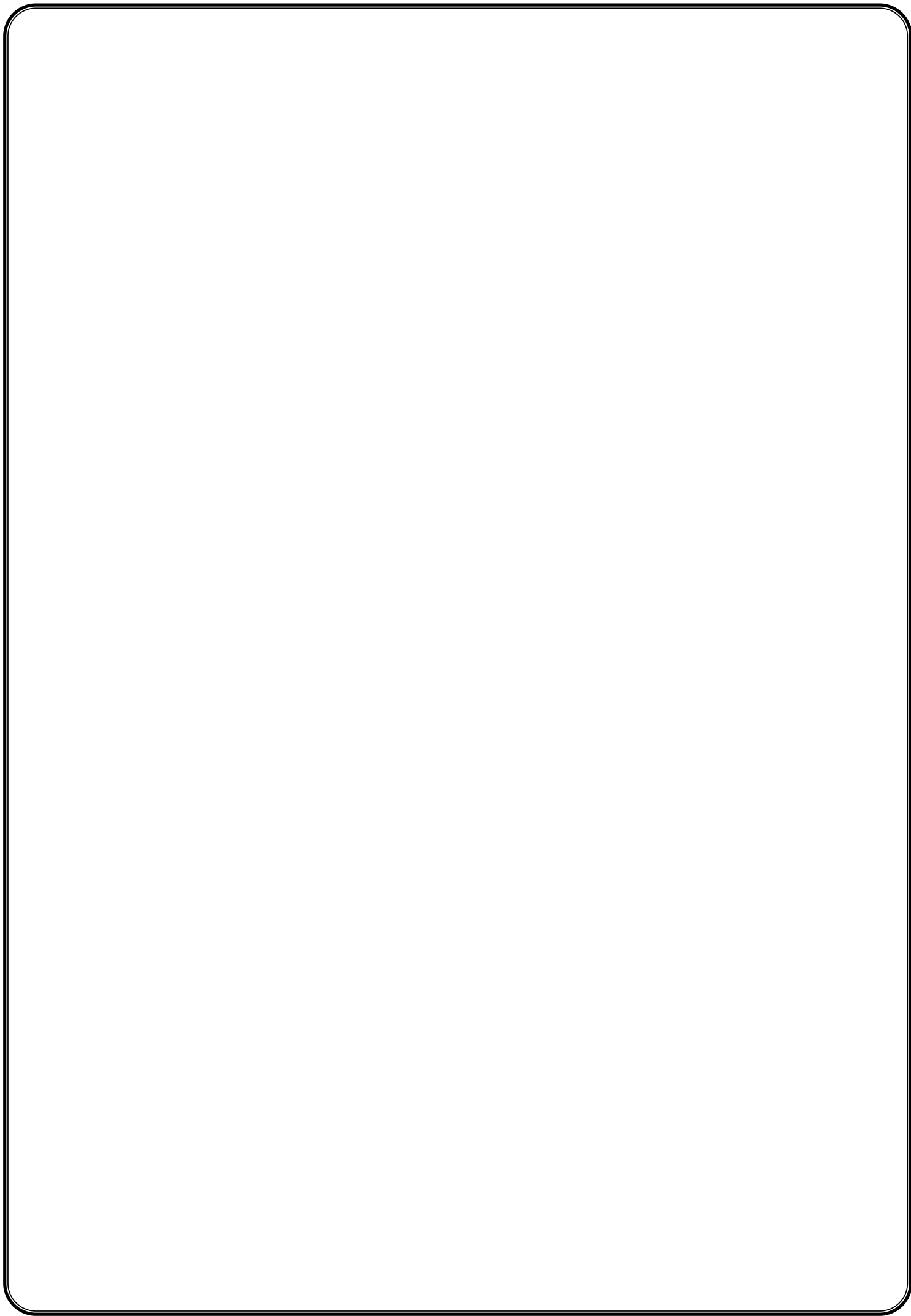
Generate Calender

reset

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				

Result :-

Program executed and output obtained successfully.



Program No.8

Form Validation

Aim :- Create a HTML registration form and to validate the form using JavaScript code.

Algorithm :-

Step 1: Start to create an html form with the required fields.

Step 2: Design the html form using CSS.

Step 3: With the help of JavaScript validate the filled data in the form.

Step 4: Enter the required data to the fields in the form.

Step 5: The data that is entered must be checked for correct form and value.

Step 6: Make sure that your code must include logic to test correctness of data.

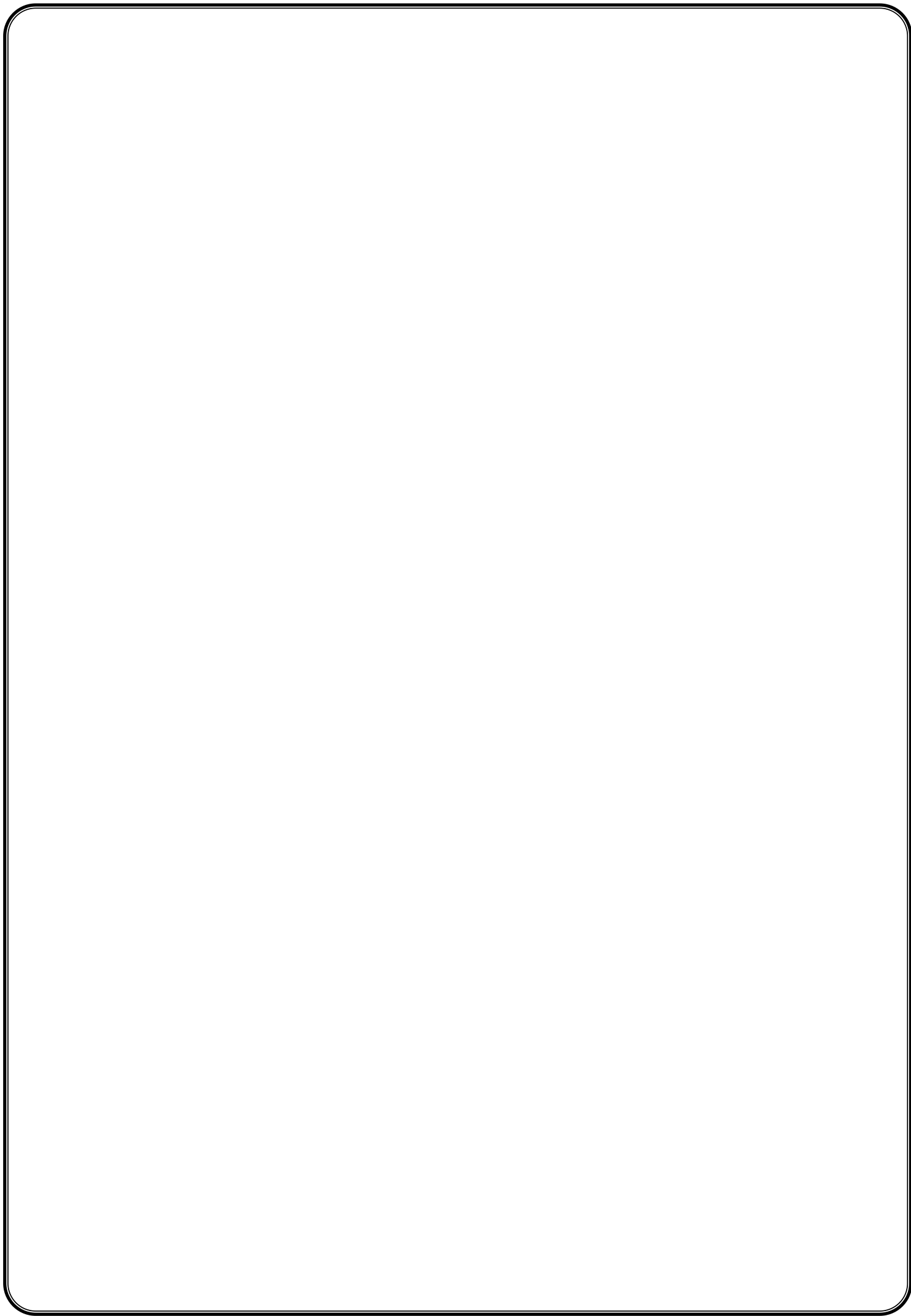
Step 7: If the entered data is invalid then an error message is raised when we click on the submit button.

Step 8: If the data is valid then accept the entered data to the database.

Step 9: Stop.

Program :-

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Form Validation</title>
</head>
<style>
table{
background-color: white;
margin-left: auto;
margin-right: auto;
margin-top:1em;
padding:1em;
box-shadow: 0 4px 10px 0 rgba(0,0,0,0.2), 0 4px 20px 0 rgba(0,0,0,0.19);
}
tr,td,th{
padding:1em;
text-align: left;
}
.center th{
text-align: center;
}
h2{
text-align: center;
margin-top: 2em;
background-color:black;
color: white;
}
</style>
<body>
<h2>FORM VALIDATION</h2>
<form name="form" action="" method="POST" onsubmit="return validateForm()">
<table>
<tr>
```

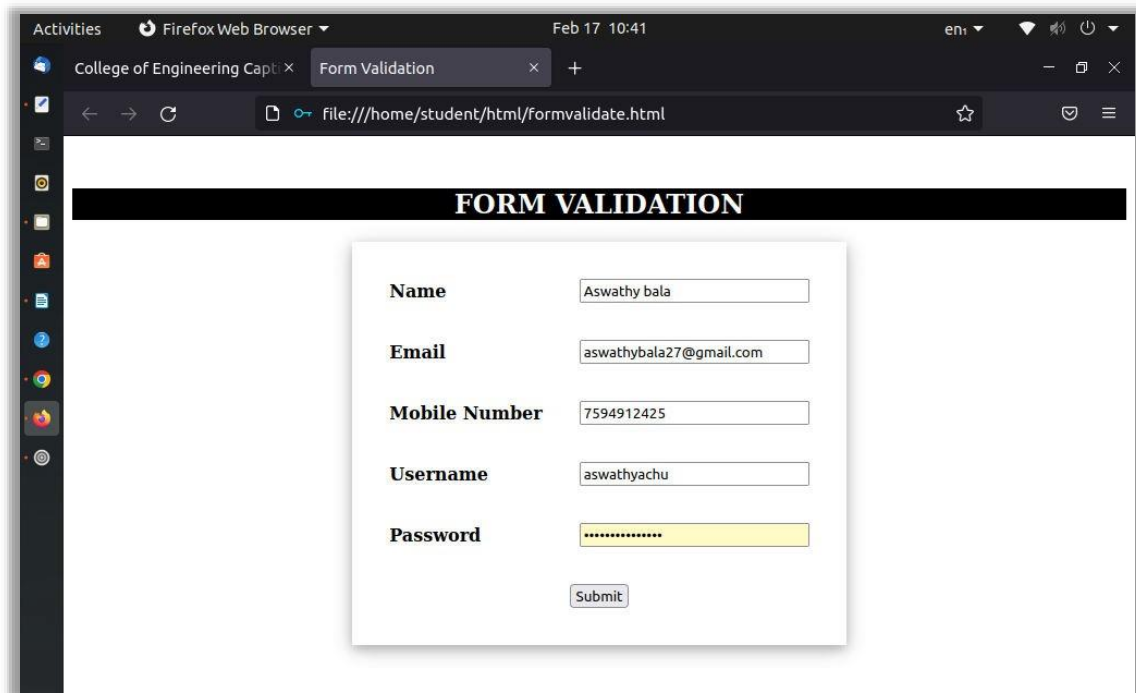


```

<th>Name</th>
<td><input type="text" name="fname"> </td>
</tr>
<tr>
<th>Email </th>
<td><input type="email" name="email"> </td>
</tr>
<tr>
<th>Mobile Number </th>
<td><input type="tel" name="mob"> </td>
</tr>
<tr>
<th>Username </th>
<td><input type="text" name="user"> </td>
</tr>
<tr>
<th>Password </th>
<td><input type="password" name="passcode"> </td>
</tr>
<tr class="center">
<th colspan="2"><input type="submit" value="Submit"></th>
</tr>
</table>
</form>
<script>
function validateForm() {
var fname = document.form.fname.value;
var mob = document.form.mob.value;
var user = document.form.user.value;
var passcode = document.form.passcode.value;
var re = /^(?=.*\d)(?=.*[!@#$%^&*])(?=.*[a-z])(?=.*[A-Z]).{8,}$/;
var ren = /[0-9!@#$%^&*]/
if (fname == "") {
alert("Enter First Name !!");
document.form.fname.focus();
return false;
}
else if(ren.test(fname)){
alert("Enter Valid Name !!");
document.form.fname.focus();
return false;
}
if (mob == "") {
alert("Enter Mobile number");
return false;
}
else if (isNaN(mob)) {
alert("Enter valid Mobile number");
return false;
}
else if (mob.length != 10) {
alert("Enter Mobile number with 10 digit");
return false;
}
}

```

Output :-



Activities Firefox Web Browser Feb 17 10:41 en

College of Engineering Capt Form Validation

file:///home/student/html/formvalidate.html

FORM VALIDATION

Name Aswathy bala

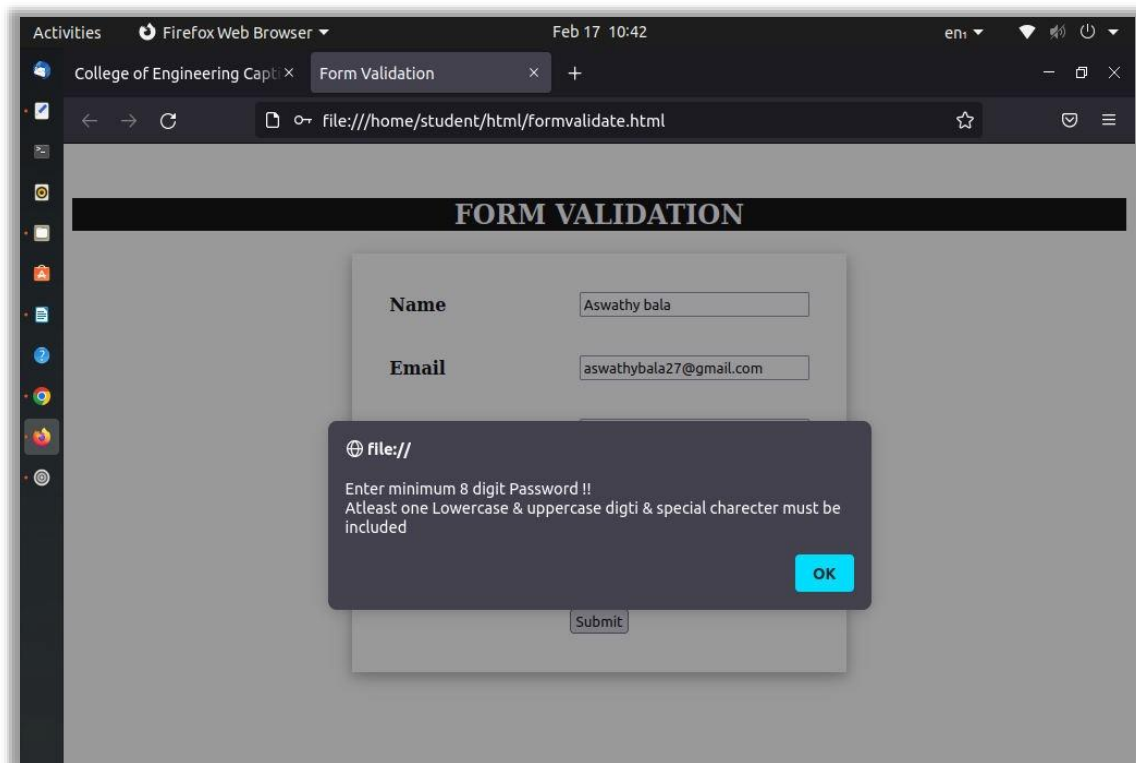
Email aswathybala27@gmail.com

Mobile Number 7594912425

Username aswathyachu

Password

Submit



Activities Firefox Web Browser Feb 17 10:42 en

College of Engineering Capt Form Validation

file:///home/student/html/formvalidate.html

FORM VALIDATION

Name Aswathy bala

Email aswathybala27@gmail.com

Mobile Number 7594912425

Username aswathyachu

Password

Submit

file:///

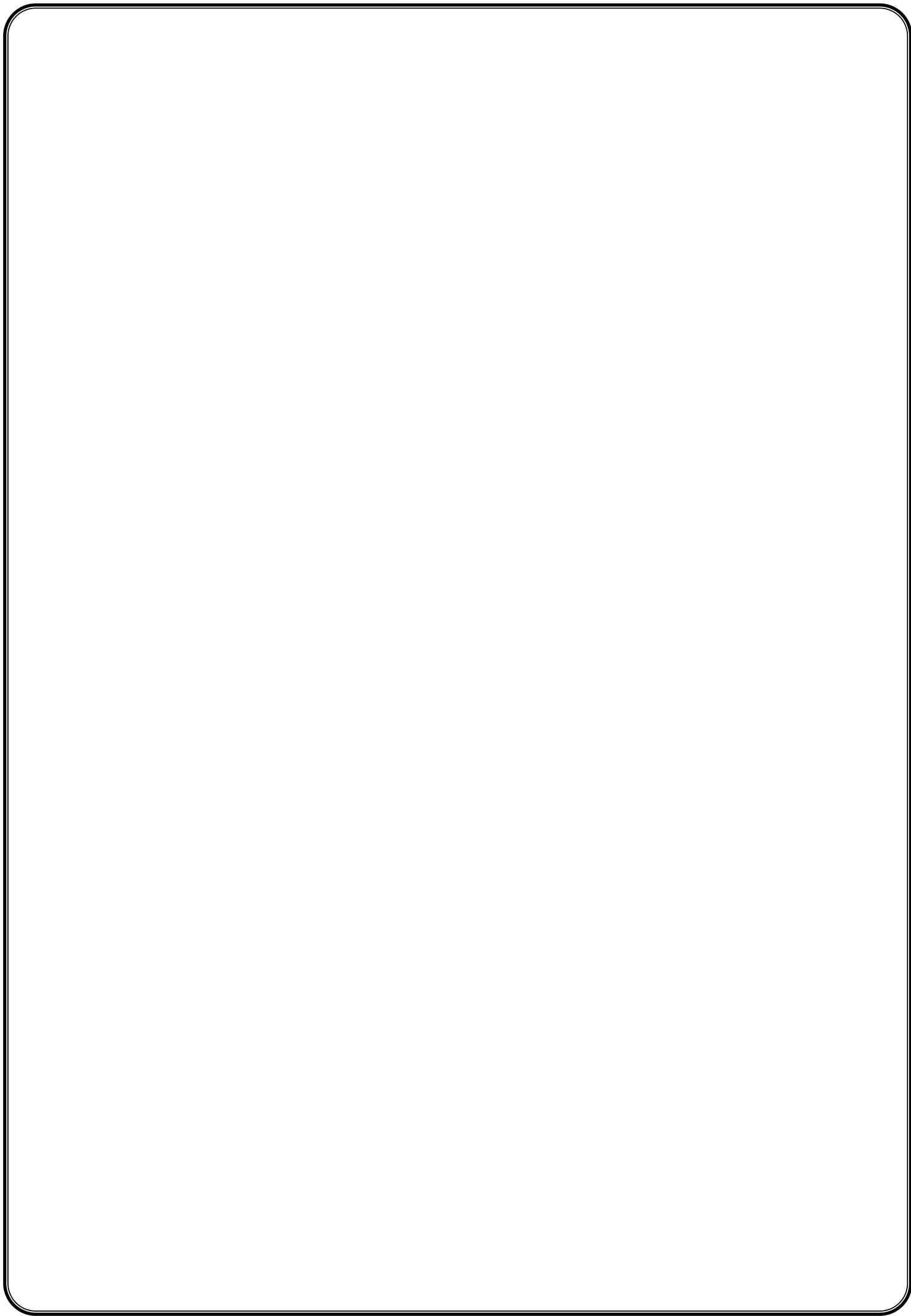
Enter minimum 8 digit Password !!
Atleast one Lowercase & uppercase digit & special character must be included

OK


```
if (user == "") {  
  
    alert("Enter User Name !!");  
    document.form.user.focus();  
    return false;  
}  
if (passcode == "" || passcode.length < 8 || !re.test(passcode)) {  
    alert("Enter minimum 8 digit Password !" + "\n" + "Atleast one Lowercase & uppercase  
    digti & special  
    charecter must be included");  
    document.form.passcode.focus();  
    return false;  
}  
}  
</script>  
</body>  
</html>
```

Result :-

Program executed and output obtained successfully.



Program No.9

Background change using JavaScript

Aim :- To create a HTML page to change the background color forProgram executed and output obtained successfully. every click of a button using JavaScript Event Handling.

Algorithm :-

Step 1 : create a html file.

Step 2 : Inside head tag, declare style tag.

Step 2.1 : create "container" named class. Set width and height100%, text align property : center and displaystyle as grid.

Step 2.2 : create another style class using button id "#btn". Set margin property : auto , padding : 1rem , border-thickness : 3pxand color : black.

Step 2.3 : Close style tag.

Step 3 : Inside body tag open a div tag , and assign classname ".container" as the newly created div's class.

Step 4 : Inside the div create a self closing input tag set id = "btn", type = "button",value = "Change Backgorund".

Step 5 : Close div tag, open a script tag.

Step 5.1 : Inside script tag create a addEventListener for the button to invoke the Function "colorChange()", when the button is clicked.

Step 5.2 : Declare the function colorChange().

Step 5.3 : Declare a variable name "color" to store the hexadecimal color code generated using math.random(), math.floor() and toString().

Step 5.4 : Set the body background style using color variable value.

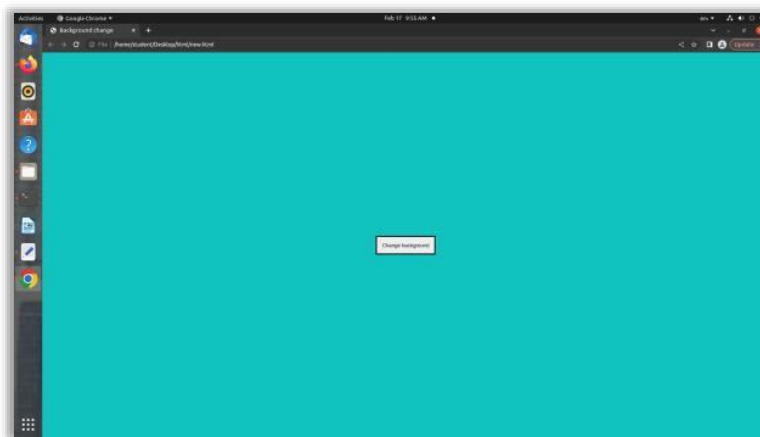
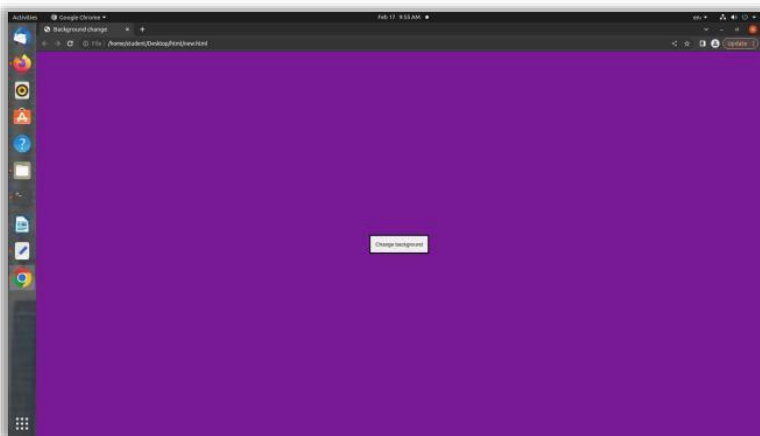
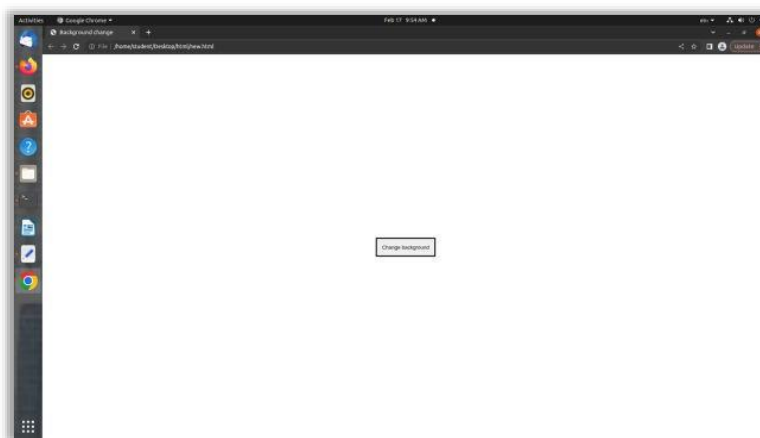
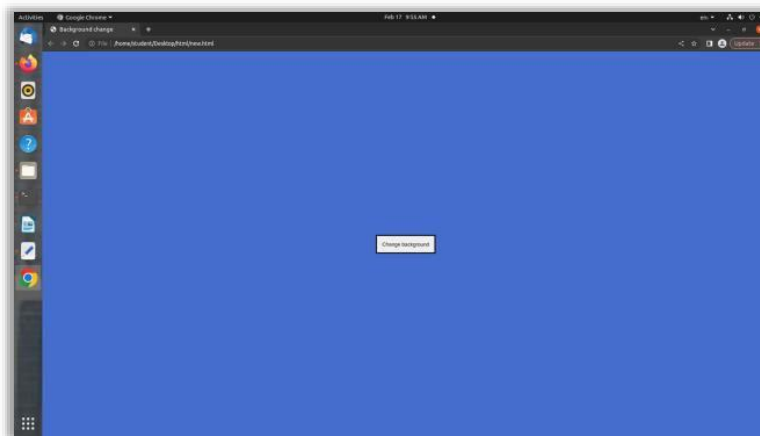
Step 6 : Close the script tag.

Step 7 : Stop.

Program :-

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Background change</title>
</head>
<style>
.container{
width:100%;
height:100%;
text-align: center;
display: grid;
}
#btn{
margin:auto;
padding:1em;
border:3px solid black;
outline:none;
}
</style>
<body>
```

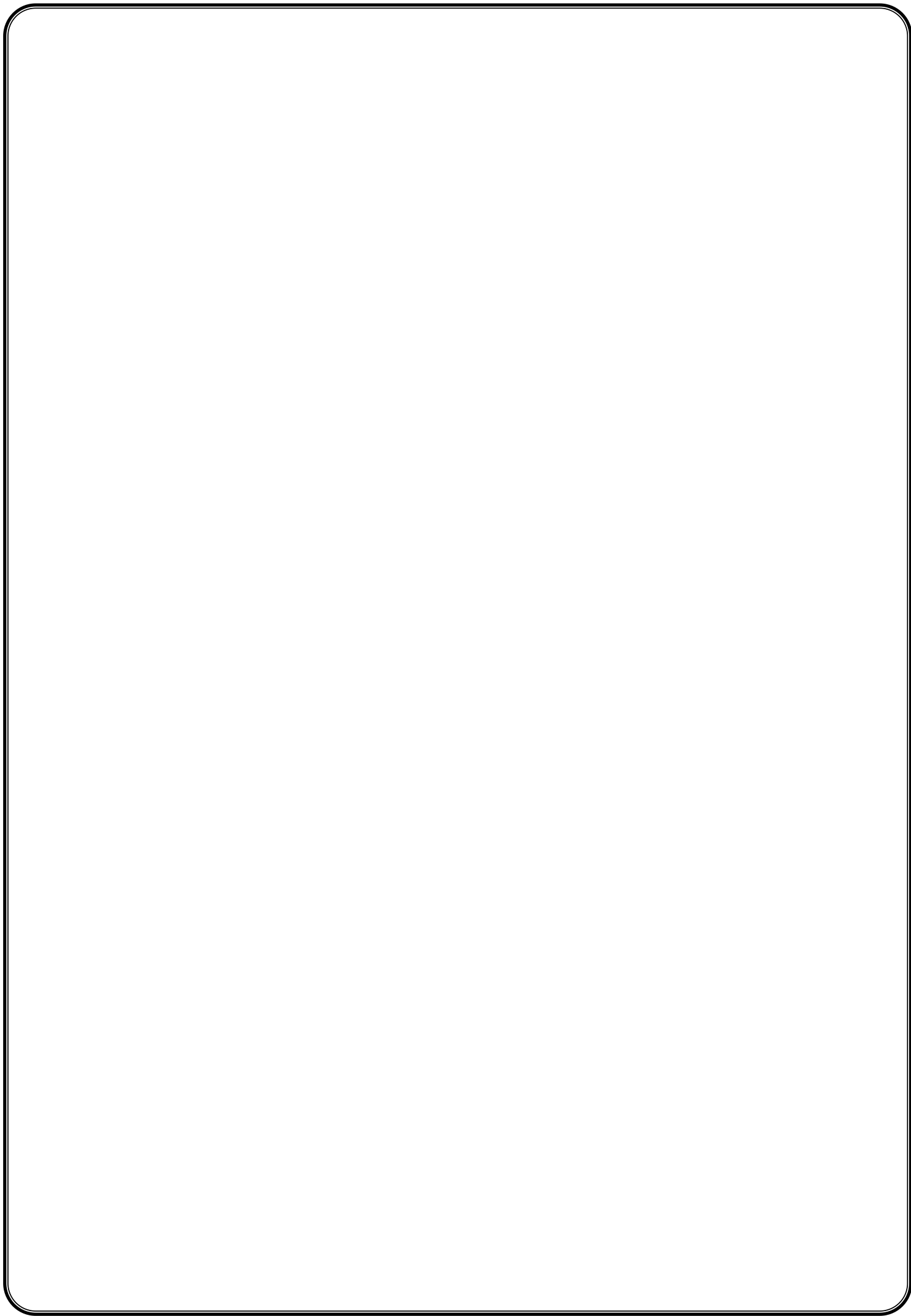
Output :-



```
<div class="container">
<input id="btn" type="button" value="Change background">
</div>
<script>
document.getElementById("btn")
function colorchange() {
console.log(Math.random() * 10000000);
var color = '#' + Math.floor(Math.random() * 10000000).toString(16);
document.body.style.background = color;
}
</script>
</body>
</html>
```

Result :-

Program executed and output obtained successfully.



Program No.10

Javascript Event Handling

Aim :- To create a HTML page to display a new image and text when the mouse comes over the existing content in the page using JavaScript Event Handling Program executed and output obtained successfully.

Algorithm :-

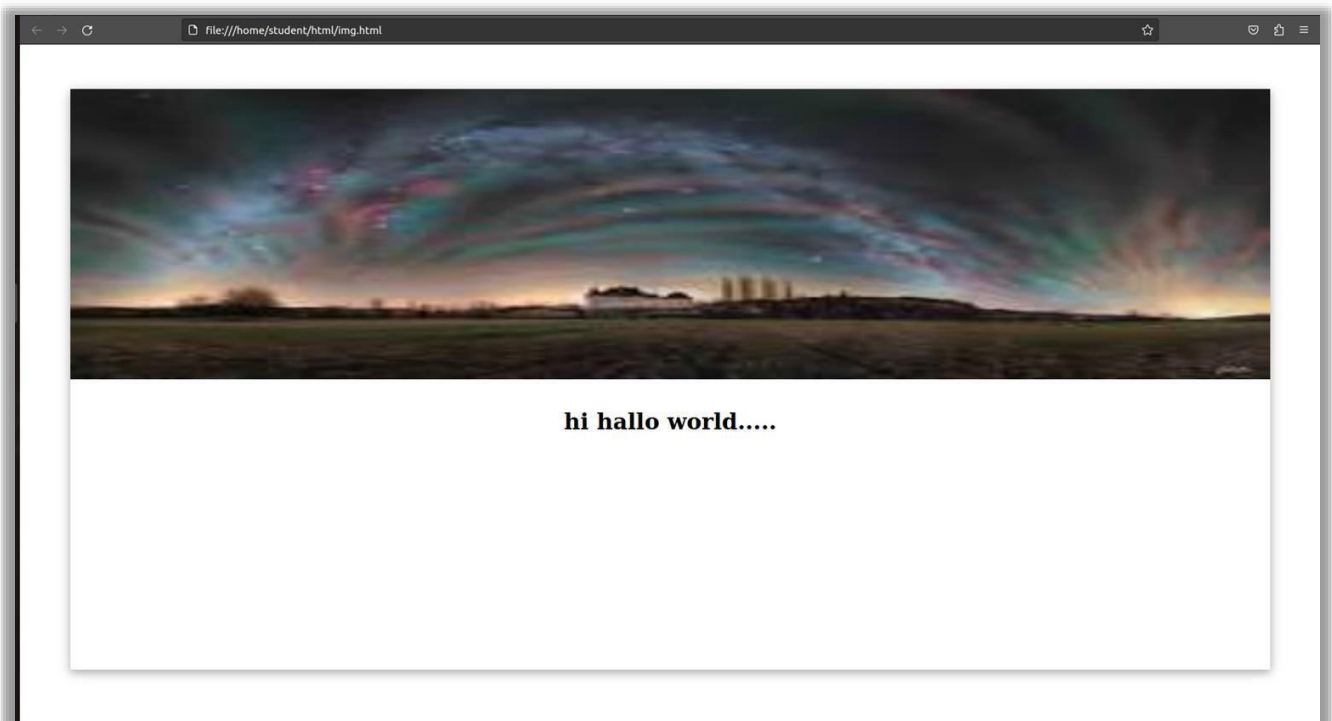
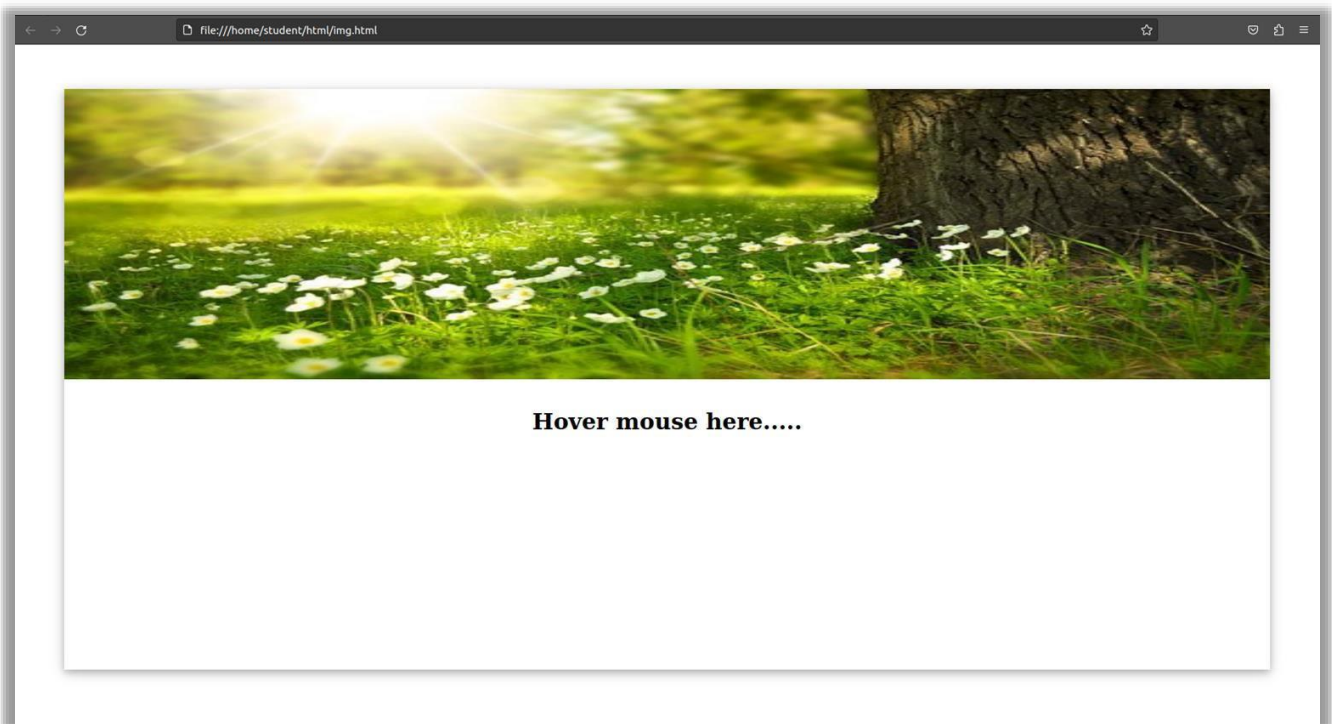
- 1.Create an HTML file with the content you want to display, such as an image and text.
2. Assign a unique ID to the HTML element containing the content.
- 3.In the JavaScript code, add an event listener to the HTML element with the ID to trigger aProgram executed and output obtained successfully.
function when the mouse hovers over the element.
- 4.In the function, change the source of the image and the text of theProgram executed and output obtained successfully. element to the new content you want to display.

Program :-

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0"><title>Image</title>
<style>
    .row {
    width: 100%;
    height: 100%;
    }
    .content {
        box-shadow: 0 4px 10px 0 rgba(0, 0, 0, 0.2), 0 4px 20px 0 rgba(0, 0, 0,
0.19);
        margin: 4em;
    height: calc(100% - 8em);
    }
    img{
    background-repeat: no-repeat;
    width: 100%;
    height:50%;
    }

    #heading1,#heading2{
    text-align: center;
    display:block;
    }
    #heading2{
    display: none;
    }
</style>
</head>
```

Output :-




```

<body>
<div class="row">
<div class="content" id="content">
  <br><br>
<div class="data">
<h1 id="heading1">Hover mouse here.....</h1>
<h1 id="heading2">hi hallo world.....</h1>
</div>
</div>
</div>
<script>

document.getElementById("content").addEventListener("mouseover",change1);
document.getElementById("content").addEventListener("mouseout",change2);
function change1(){
  document.getElementById('image').src = "parr.jpg";
  document.getElementById('heading1').style.display="none";
  document.getElementById('heading2').style.display="block";
}
function change2(){
  document.getElementById('image').src = "fli.jpg";
  document.getElementById('heading1').style.display="block";
  document.getElementById('heading2').style.display="none";
}
</script>
</body>
</html>

```

Result :-

Program executed and output obtained successfully.