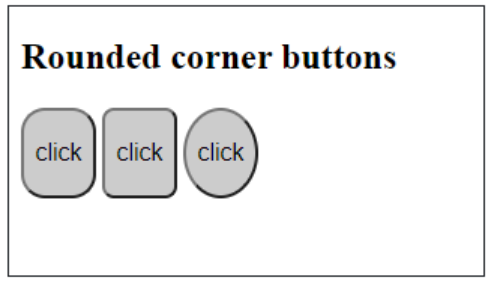
Practical - 8

Working with css – ii

1. Create a webpage which implements CSS to generate the following output [hint use border radius property].



<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 1</title>

    <style>

      .container {

        border: 1px solid black;

        width: 500px;

        margin: auto;

        padding: 20px;

      }

      .box\_container {

        display: flex;

        gap: 10px;

      }

      .box1,

      .box2,

      .box3 {

        width: 100px;

        height: 100px;

        display: flex;

        justify-content: center;

        align-items: center;

        border: 2px solid gray;

        border-radius: 15px;

      }

      .box2{

        border-radius: 8px;

      }

      .box3{

        border-radius: 50%;

      }

    </style>

  </head>

  <body>

    <div class="container">

      <h2>Rounded corner buttons</h2>

      <div class="box\_container">

        <div class="box1">click</div>

        <div class="box2">click</div>

        <div class="box3">click</div>

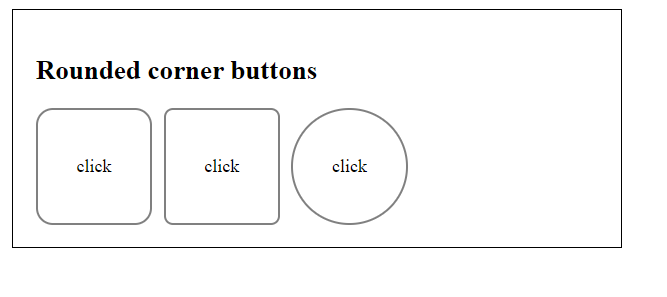
      </div>

    </div>

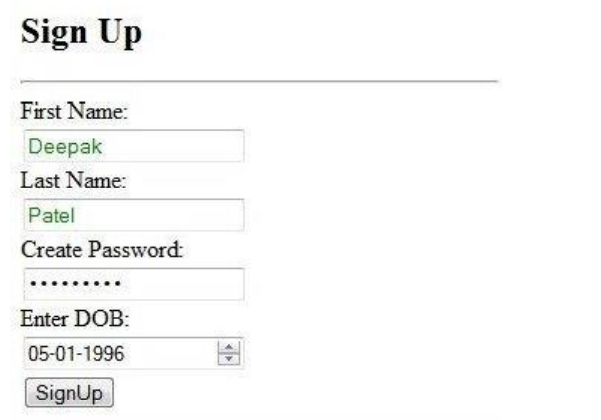
  </body>

</html>

**Output :**



1. Create following form and write CSS to set green color text for the textboxes.



<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 2</title>

    <style>

      h1 {

        border-bottom: 1px solid;

        padding-bottom: 10px;

      }

      input[type="text"] {

        color: green;

      }

      .fields {

        display: flex;

        flex-direction: column;

        width: 300px;

        margin-bottom: 8px;

      }

    </style>

  </head>

  <body>

    <div>

      <h1>Sign Up</h1>

      <form action="">

        <div class="fields">

          <label for="fname">First name : </label>

          <input type="text" id="fname" />

        </div>

        <div class="fields">

          <label for="lname">Last name : </label>

          <input type="text" id="lname" />

        </div>

        <div class="fields">

          <label for="pwd">Create Password : </label>

          <input type="password" id="pwd" />

        </div>

        <div class="fields">

          <label for="dob">Enter DOB : </label>

          <input type="date" id="dob" />

        </div>

        <div >

          <input type="submit" value="Sign up" />

        </div>

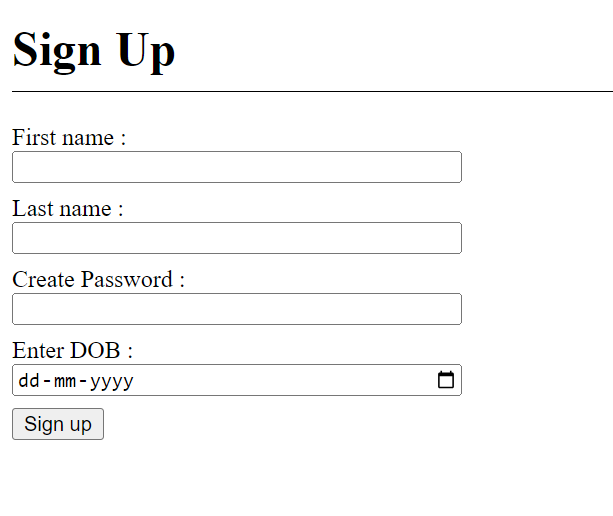
      </form>

    </div>

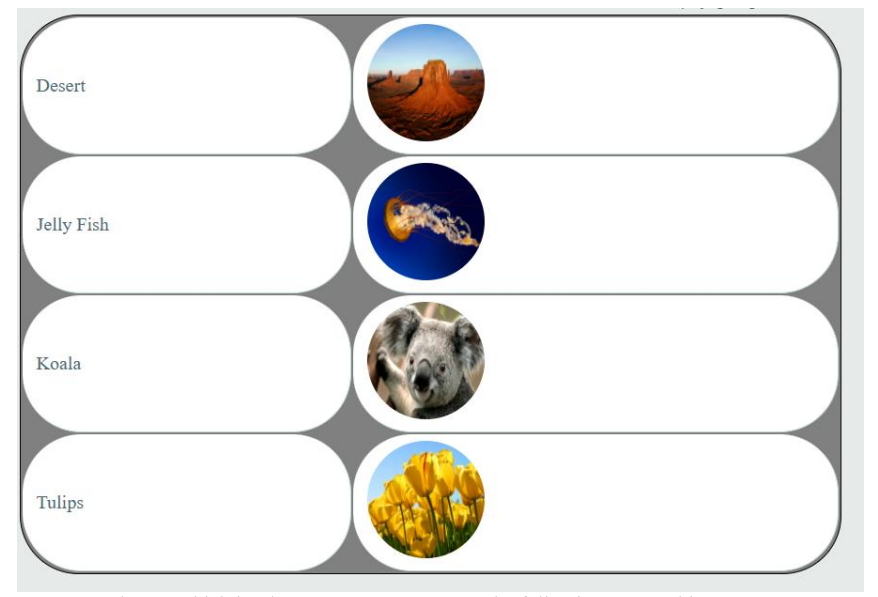
  </body>

</html>

Output :



1. Create a webpage which implements CSS to generate the following output [hint use border radius property for td and img]



<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 3</title>

    <style>

      table {

        background-color: gray;

        width: 500px;

        border-radius: 30px;

      }

      td {

        border-radius: 30px;

        background-color: white;

      }

      img {

        border-radius: 50%;

        height: 80px;

        width: 80px;

      }

    </style>

  </head>

  <body>

    <div>

      <table>

        <tr>

          <td>Desert</td>

          <td><img src="./desert.jpg" alt="desert" /></td>

        </tr>

        <tr>

          <td>Jelly Fish</td>

          <td><img src="./jelly fish.jpg" alt="jellyfish" /></td>

        </tr>

        <tr>

          <td>Koala</td>

          <td><img src="./koala.jpg" alt="koala" /></td>

        </tr>

        <tr>

          <td>Tulips</td>

          <td><img src="./tulips.jpg" alt="tulips" /></td>

        </tr>

      </table>

    </div>

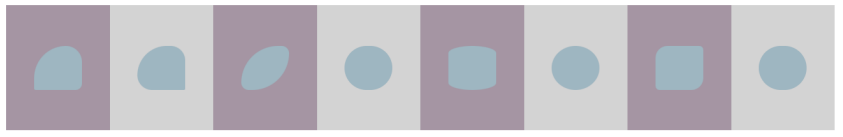
  </body>

</html>

**Output :**



1. Create a webpage which implements CSS to generate the following output [hint use border radius property]



<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 4</title>

    <style>

      .Container {

        display: flex;

      }

      .innerContainer {

        width: 200px;

        height: 200px;

        display: flex;

        justify-content: center;

        align-items: center;

        padding: 10px;

      }

      .innerContainer:nth-child(even) {

        background: #d3d3d3;

      }

      .innerContainer:nth-child(odd) {

        background: #a595a3;

      }

      .box1,

      .box2,

      .box3,

      .box4,

      .box5,

      .box6,

      .box7,

      .box8 {

        width: 100px;

        height: 100px;

        background-color: #9eb6c1;

      }

      .box1 {

        border-radius: 90px 30px 15px 5px;

      }

      .box2 {

        border-radius: 90px 40px 5px 40px;

      }

      .box3 {

        border-radius: 125px 20px 125px 20px;

      }

      .box7 {

        border-radius: 25px 8px 25px 8px;

      }

      .box5 {

        border-radius: 50%/10%;

      }

      .box4,

      .box6,

      .box8 {

        border-radius: 50%;

      }

    </style>

  </head>

  <body>

    <div class="Container">

      <div class="innerContainer">

        <div class="box1"></div>

      </div>

      <div class="innerContainer">

        <div class="box2"></div>

      </div>

      <div class="innerContainer">

        <div class="box3"></div>

      </div>

      <div class="innerContainer">

        <div class="box4"></div>

      </div>

      <div class="innerContainer">

        <div class="box5"></div>

      </div>

      <div class="innerContainer">

        <div class="box6"></div>

      </div>

      <div class="innerContainer">

        <div class="box7"></div>

      </div>

      <div class="innerContainer">

        <div class="box8"></div>

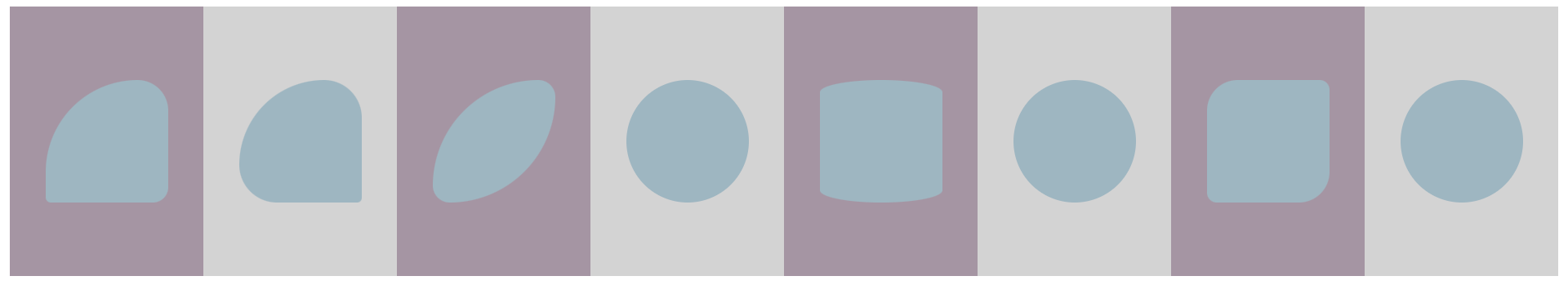
      </div>

    </div>

  </body>

</html>

**Output :**



1. Create a webpage which implements border style for each side. The top and right borders must be dashed, the bottom and left borders must be dotted. All sides must also have "2px" and "purple" values.

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 5</title>

    <style>

      #box1 {

        height: 250px;

        width: 250px;

        border-top: 2px purple dashed;

        border-right: 2px purple dashed;

        border-left: 2px purple dotted;

        border-bottom: 2px purple dotted;

      }

    </style>

  </head>

  <body>

    <div id="box1"></div>

  </body>

</html>

**Output :**



1. Create a webpage which implements mixed border and image border.

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 6</title>

    <style>

      #b1 {

        margin: 50px;

        height: 280px;

        width: 280px;

        border-top: 2px purple dashed;

        border-right: 2px purple solid;

        border-left: 2px purple groove;

        border-bottom: 2px purple dotted;

      }

      #imgbor {

        margin: 50px;

        height: 280px;

        width: 280px;

        border: 10px solid transparent;

        border-image: url(./desert.jpg) 30 round;

      }

    </style>

  </head>

  <body>

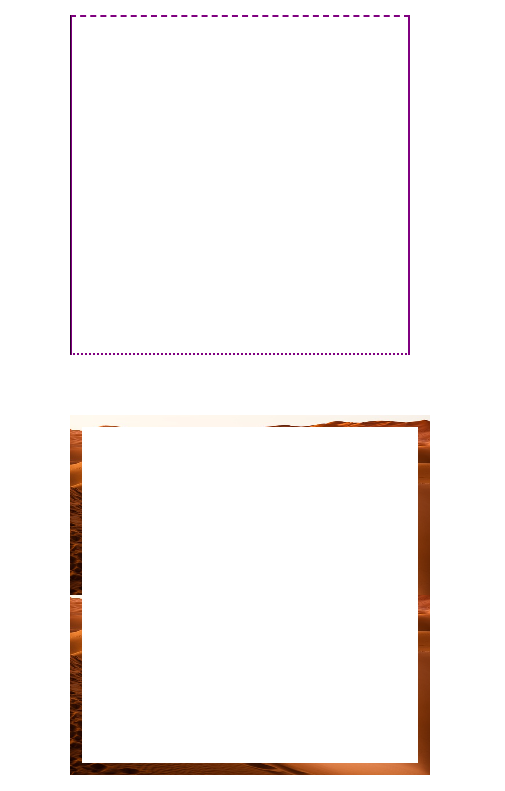
    <div id="b1"></div>

    <div id="imgbor"></div>

  </body>

</html>

**Output :**



7. Use uppercase letters for the text in the <h2> tag, and capitalized letters for the text in the

<span> tag. Indent the first line of the <div> tag with 50px. [hint use text transform, text

indent ]

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 7</title>

    <style>

      h2 {

        text-transform: uppercase;

      }

      span {

        text-transform: capitalize;

      }

      div {

        text-indent: 50px;

      }

    </style>

  </head>

  <body>

    <h2>

      Adjust the columns and conditions according to your actual table

      structures and relationships.

    </h2>

    <span

      >It's important to note that the choice of JOIN type depends on your

      specific requirements.</span

    >

    <div>

      The specific relationships between the tables determine how you join them

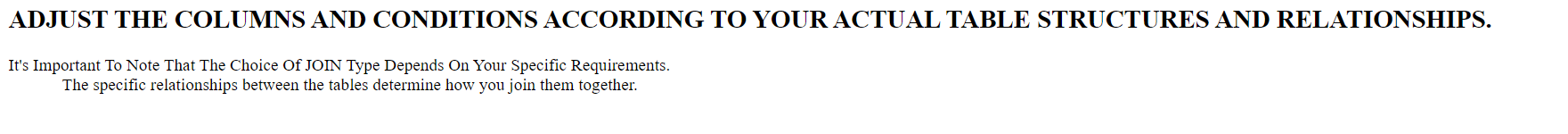
      together.

    </div>

  </body>

</html>

**Output :**



8. Create a webpage to implement opacity. Here use three heading tag h1, h2, h3 set its background color to red and different opacity values. So after implementation what we can see is, the heading 1 is not visible at all. In heading 2 the content is quite blurred but visible while in heading 3 is clearly visible.

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 8</title>

    <style>

      h1 {

        background-color: red;

        opacity: 0;

      }

      h2 {

        background-color: red;

        opacity: 0.5;

      }

      h3 {

        background-color: red;

        opacity: 1;

      }

    </style>

  </head>

  <body>

    <h1>

      Lorem ipsum dolor sit amet consectetur adipisicing elit. Tempore,

      obcaecati?

    </h1>

    <h2>Lorem ipsum dolor sit amet consectetur.</h2>

    <h3>

      Lorem ipsum, dolor sit amet consectetur adipisicing elit. At vel eligendi

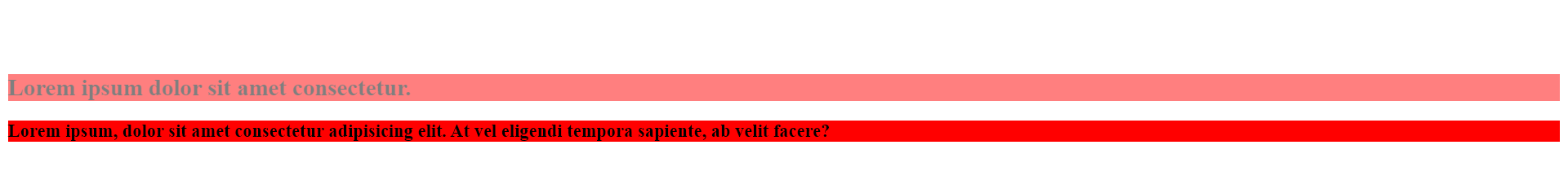
      tempora sapiente, ab velit facere?

    </h3>

  </body>

</html>

**Output :**



9. Create a webpage to implement the multi-column model. [hint: use the following properties column-count, column-width, column-gap, column-span]

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <style>

      .multicolumn-container {

        column-count: 3; /\* Number of columns \*/

        column-width: 200px; /\* Width of each column \*/

        column-gap: 20px; /\* Gap between columns \*/

      }

      .column-item {

        margin-bottom: 20px;

        padding: 10px;

        border: 1px solid #ccc;

      }

    </style>

    <title>Multi-Column Layout</title>

  </head>

  <body>

    <div class="multicolumn-container">

      <div class="column-item">

        <h2>Column 1</h2>

        <p>This is the content of column 1.</p>

      </div>

      <div class="column-item">

        <h2>Column 2</h2>

        <p>This is the content of column 2.</p>

      </div>

      <div class="column-item">

        <h2>Column 3</h2>

        <p>This is the content of column 3.</p>

      </div>

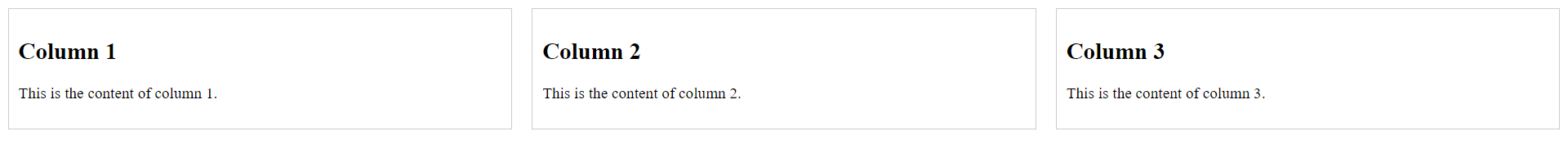
      <!-- Add more column items as needed -->

    </div>

  </body>

</html>

**Output :**



10. Create a webpage to implement the template-layout model.

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Program 10</title>

    <style>

      .Container {

        display: grid;

        gap: 4px;

        grid-template-areas: "head head head head head" "nav con con con oth" "foot foot foot foot foot";

      }

      .Container > div {

        background-color: black;

        color: white;

        padding: 10px;

        text-align: center;

      }

      #Header {

        grid-area: head;

      }

      #Menu {

        grid-area: nav;

        min-height: 300px;

      }

      #Content {

        grid-area: con;

        min-height: 300px;

      }

      #Other {

        grid-area: oth;

        min-height: 300px;

      }

      #Footer {

        grid-area: foot;

      }

    </style>

  </head>

  <body>

    <div class="Container">

      <div id="Header">Header</div>

      <div id="Menu">Menu</div>

      <div id="Content">Content</div>

      <div id="Other">Other</div>

      <div id="Footer">Footer</div>

    </div>

  </body>

</html>

**Output :**

