CSS-HTML Exam —-4bhrs

1. Design a form with input fields for name, email, and message.

**Html:**

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <title>fields</title>
7. <link rel="stylesheet" href="form.css">
8. </head>
9. <body>
10. <div class="total-page">
11. <form class="form">
12. <div class="content">
13. <label for="name">Name:</label>
14. <input type="text" placeholder="type here" id="name" name="name" required>
15. </div>
16. <div class="content">
17. <label for="email">Email:</label>
18. <input type="text" placeholder="type here" id="email" name="email">
19. </div>
20. <div class="content-page">
21. <label for="message">Message:</label>
22. <textarea name="message" id="message"></textarea>
23. </div>
24. <div class="button">
25. <input type="submit" value="submit">
26. </div>
27. </form>
28. </div>
29. </body>
30. </html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.total-page{

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

.total-page .form{

    border: 2px solid rgb(237, 241, 233);

    padding: 60px;

    background-color:rgb(226, 171, 190);

    border-radius: 10px;

}

.total-page .form .content input {

   width: 100%;

   padding: 5px;

   border: 2px solid whitesmoke;

   border-radius: 2px;

   margin-bottom: 10px;

}

.total-page .form  .button input{

    width: 100%;

    padding: 5px;

    border: none;

    border-radius: 5px;

    margin-bottom: 8px;

    margin-top: 5px;

    color:black;

    background-color: aqua;

  }

.total-page .form  .content-page textarea{

    width: 100%;

  padding: 5px;

  border: 2px solid whitesmoke;

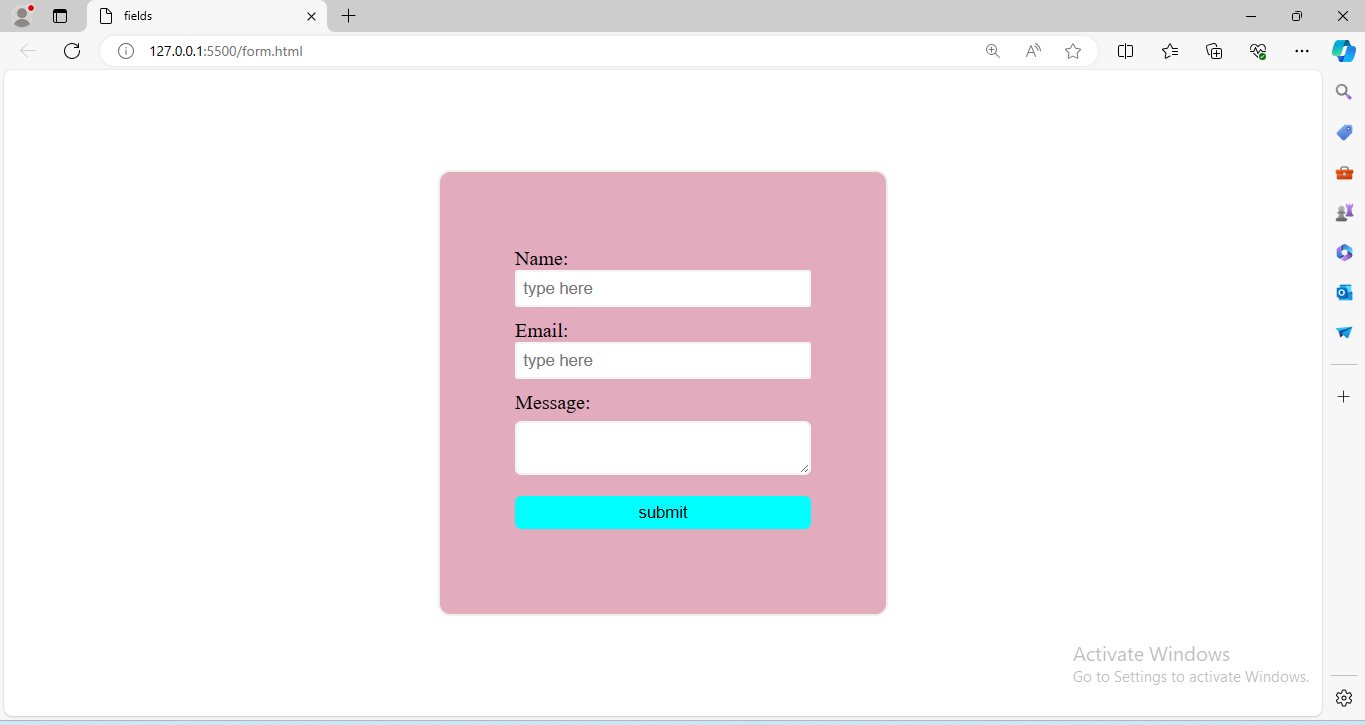
  border-radius: 5px;

  margin-bottom: 8px;

  margin-top: 5px;

}

**Output:**

****2. Create a three-column layout using CSS grid or flexbox

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>columns</title>

    <link rel="stylesheet" href="column.css">

</head>

<body>

    <div class="total-page">

        <div class="column">

            <h1>column 1</h1>

            <p>HTML is the standard markup language for Web pages</p>

        </div>

        <div class="column">

            <h2>column 2</h2>

            <p>A browser does not display the HTML tags, but uses them to determine how to display the document</p>

        </div>

        <div class="column">

            <h3>column 3</h3>

            <p>Since the early days of the World Wide Web, there have been many versions of HTML</p>

        </div>

    </div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.total-page{

    display: flex;

    justify-content:space-between;

    align-items: center;

    height: 100vh;

}

.total-page .column{

    border: 2px solid black;

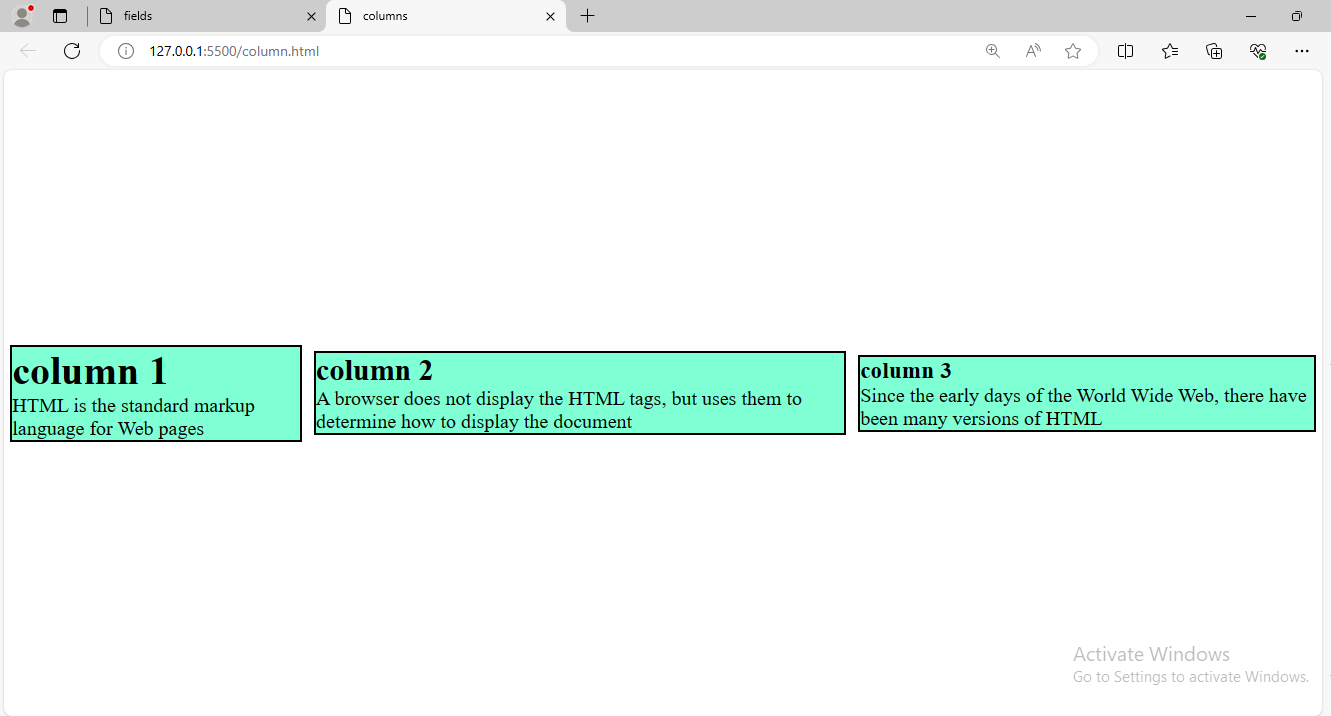
    margin-right: 5px;

    margin-left: 5px;

    background-color: aquamarine;

}

**Output:**

****

1. Make a responsive website that adjusts its layout for different screen sizes.

**Html:**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>table</title>

    <link rel="stylesheet" href="responsive.css" />

  </head>

  <body>

    <div class="table-container">

      <table>

        <tr>

          <th>name</th>

          <th>qualification</th>

          <th>percentage</th>

        </tr>

        <tr>

          <td>sony</td>

          <td>BTech</td>

          <td>72%</td>

        </tr>

        <tr>

          <td>Raju</td>

          <td>Degree</td>

          <td>67%</td>

        </tr>

        <tr>

          <td>swetha</td>

          <td>BTech</td>

          <td>68%</td>

        </tr>

      </table>

    </div>

  </body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .table-container {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

  }

  table,

  th,

  td {

    border: 2px solid lightpink;

    border-collapse: collapse;

    padding: 10px;

  }

  table tr:nth-child(even) {

    background-color: rgb(2, 186, 146);

    color: white;

  }

  /\* Responsive styles \*/

  @media screen and (max-width: 1024px) {

    .table-container {

      background-color: rgb(255, 211, 170);

    }

    table,

    th,

    td {

      padding: 8px;

    }

  }

  @media screen and (max-width: 768px) {

    .table-container {

      background-color: rgb(255, 227, 204);

    }

    table,

    th,

    td {

      padding: 6px;

    }

  }

  @media screen and (max-width: 480px) {

    .table-container {

      background-color: rgb(255, 204, 204);

    }

    table,

    th,

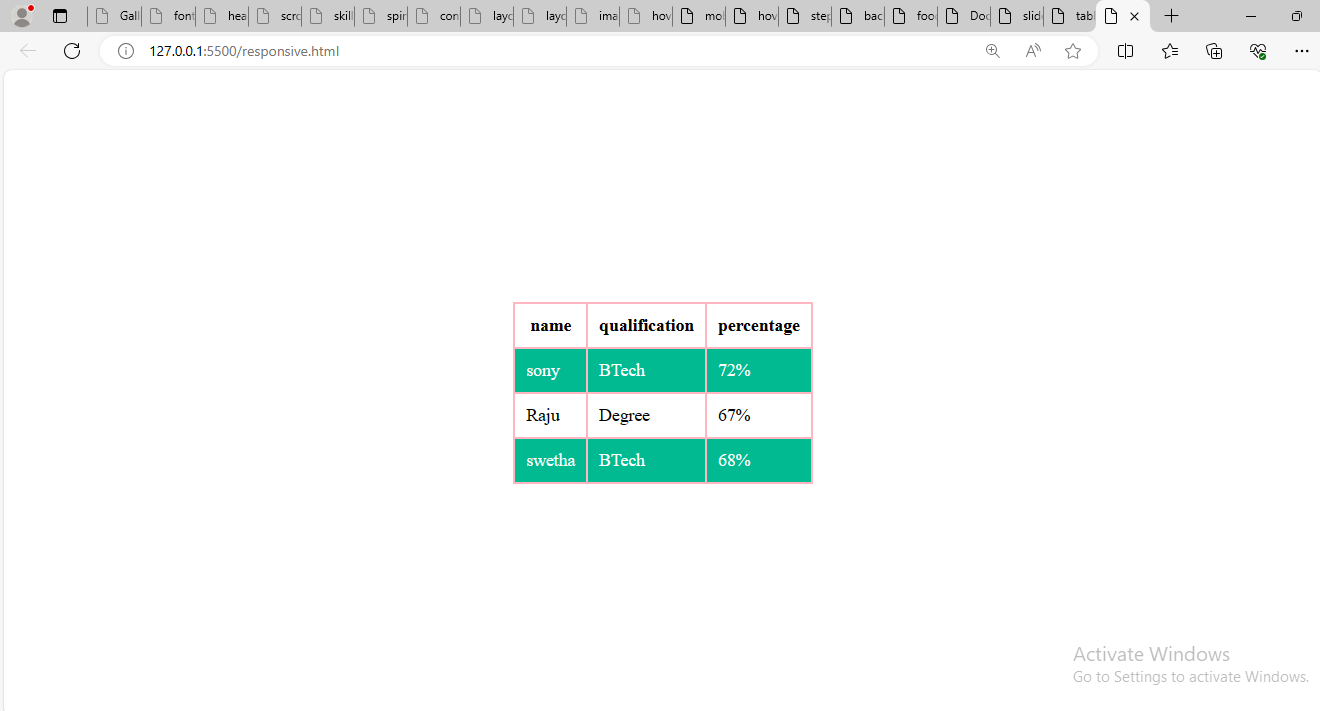
    td {

      padding: 4px;

    }

  }

**Output:**

****

1. Style a button to change its appearance when hovered over.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>button</title>

    <link rel="stylesheet" href="hover.css">

</head>

<body>

    <div class="total-page">

        <form class="fields">

        <div class="form">

    <label for="name">Name:</label>

    <input type="text" placeholder="type here" id="name" name="name" required>

</div>

    <button class="button">submit</button>

</form>

</div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.total-page{

    display: flex;

    align-items: center;

    justify-content: center;

    height: 100vh;

}

.total-page .fields{

    border: 2px solid black;

    border-radius: 5px;

    padding: 50px;

}

.total-page .fields .form input{

    width: 100%;

    padding: 10px;

    border: 2px solid rgb(22, 160, 202);

    border-radius: 5px;

    margin-bottom: 5px;

}

.total-page .fields .button{

width: 100%;

padding: 5px;

border: none;

border-radius: 5px;

margin-bottom: 8px;

margin-top: 5px;

color:black;

background-color: aqua;

}

.total-page .fields .button:hover{

    color: red;

    padding: 10px;

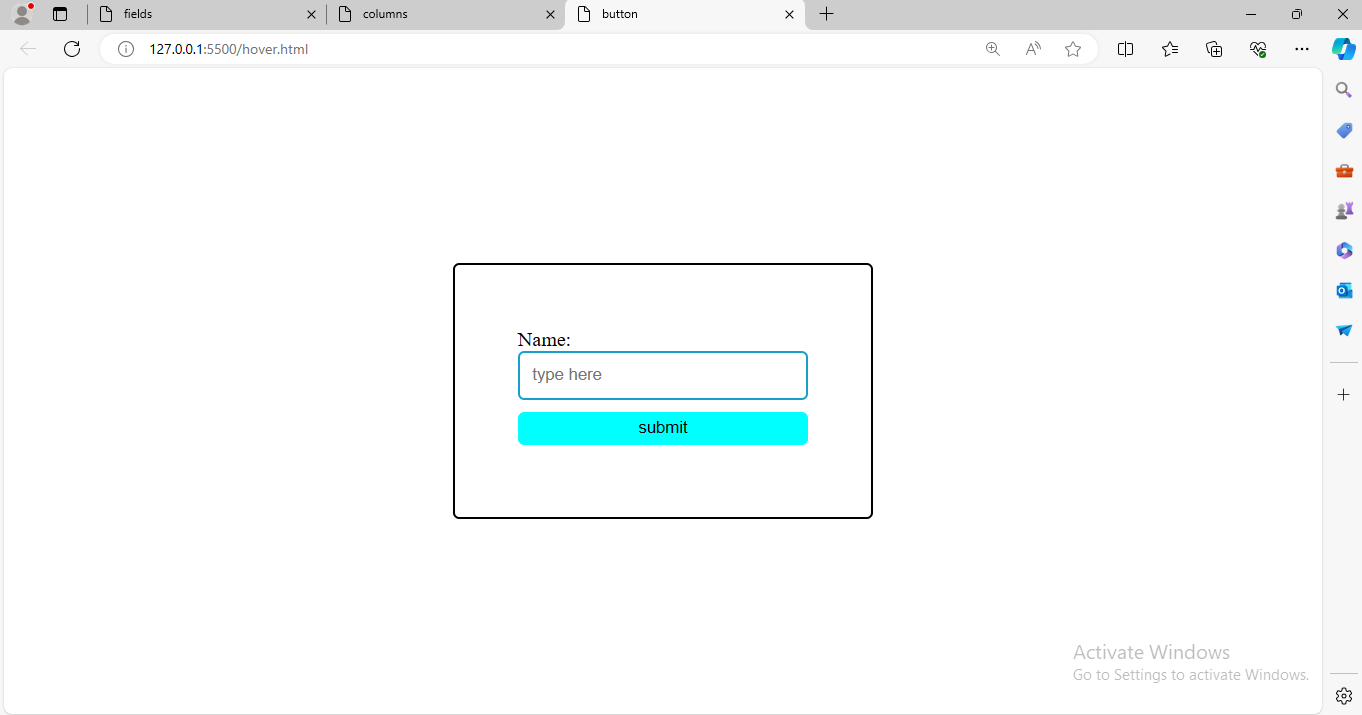
    background-color: black;

    cursor: pointer;

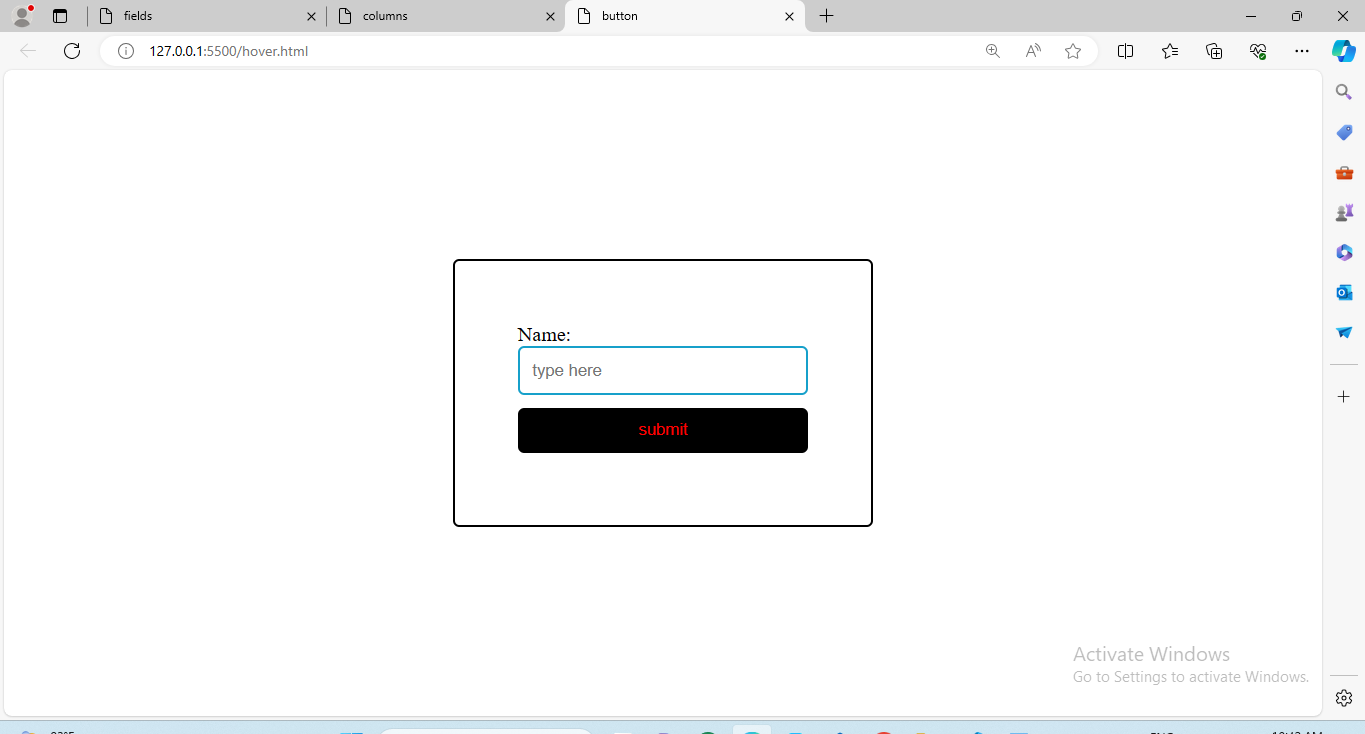
}

**Output:**

**Without hover:**

****

**With hover:**

****

1. Create a dropdown menu using HTML and CSS.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Document</title>

    <link rel="stylesheet" href="drop.css">

</head>

<body>

    <div class="dropdown">

    <select>

        <option value="select the place">select the place</option>

        <option value="kerela">kerela</option>

        <option value="mumbai">mumbai</option>

        <option value="shimla">shimla</option>

        <option value="bangalore">bangalore</option>

        <option value="hyderabad">hyderabad</option>

        <option value="chennai">chennai</option>

        <option value="telangana">telangana</option>

</select>

</div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.dropdown{

   display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

.dropdown select{

    border: 3px solid rgb(153, 59, 172);

    border-radius: 10px;

    padding: 5px;

    width: 50%;

    color: black;

    cursor: pointer;

}

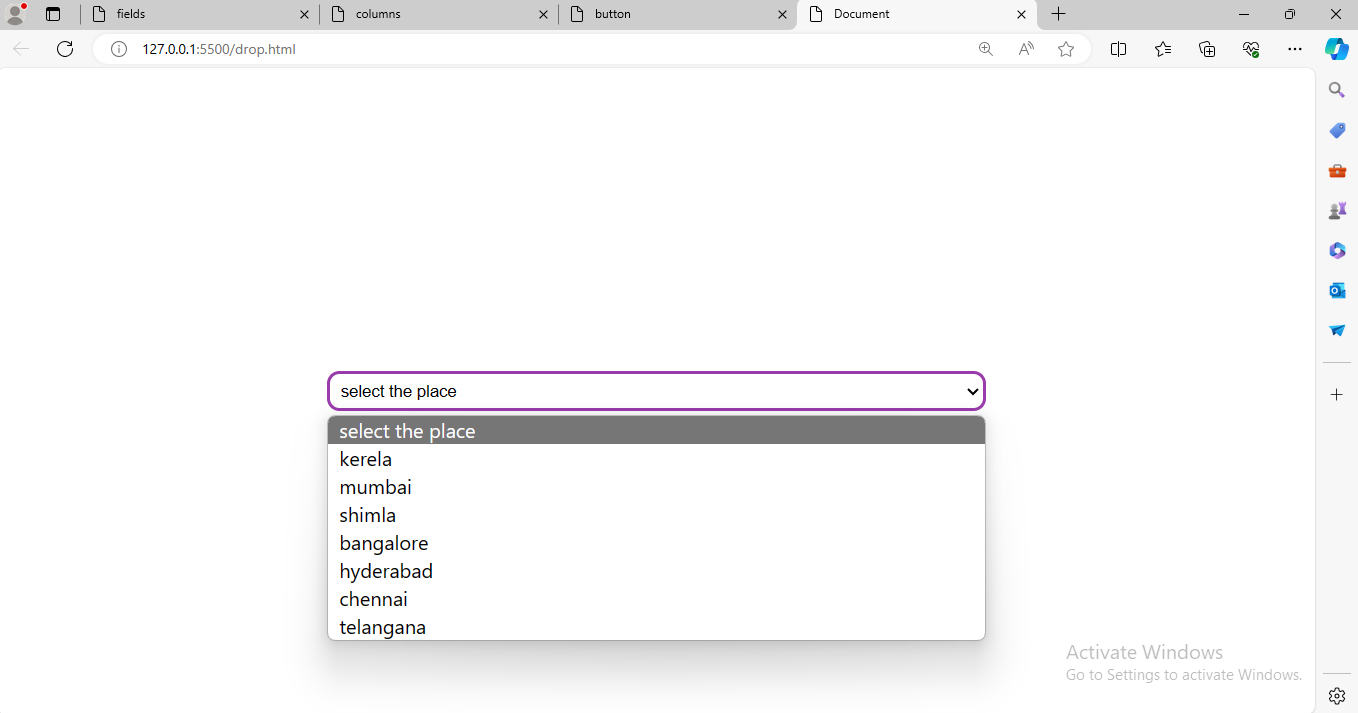
.dropdown select option{

    font-size: medium;

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

**Output:**

****

1. Build a simple slideshow using HTML, CSS, and JavaScript.
2. Design a pricing table with multiple plans using HTML and CSS.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>price table</title>

    <link rel="stylesheet" href="subscribe.css">

</head>

<body>

    <div class="subscribe">

        <div class="amount">

            <h1>Gold subscription</h1>

            <div>4K · Dolby 5.1 · Telugu + Tamil Movies & Web series · 1 Year</div>

            <footer>

                <p>INR 899 / year

                    </p>

                    <button>Subscribe</button>

            </footer>

        </div>

        <div class="amount">

        <h1>Annual Premium</h1>

        <div>Full HD (1080p) · 5.1 · Telugu Movies & Web series · 1 Year</div>

        <footer>

            <p>INR 699 / year

               </p>

                <button>Subscribe</button>

            </div>

            <div class="amount">

                <h1>Quarterly Mobile</h1>

                <div>HD (720p) · Stereo · Telugu Movies & Web series · 3 Months · Stream on Mobile only</div>

                <footer>

                    <p>INR 99 / 3 months</p>

                        <button>Subscribe</button>

                    </div>

                    <div class="amount">

                        <h1>Annual</h1>

                        <div>Full HD (1080p) · Stereo · Telugu Movies & Web series · 1 Year</div>

                        <footer>

                            <p>INR 399 / year

                                </p>

                                <button>Subscribe</button>

                            </div>

        </footer>

    </div>

</body>

</html>

**Css:**

\*{

    margin:0px;

    padding: 0px;

    box-sizing: border-box;

}

.subscribe{

    display: flex ;

    justify-content: space-between;

    align-items: center;

    height: 100vh;

}

.subscribe .amount{

    border: 3px solid wheat;

    background-color: rgb(94, 216, 224);

    width: 50%;

    padding: 50px;

    margin: 10px;

    border-radius: 10px;

    cursor: pointer;

}

.amount:hover{

    color: rgb(177, 85, 10);

}

.subscribe button{

    margin-top: 20px;

    background-color: rgb(238, 107, 59);

    cursor: pointer;

}

.subscribe button:hover{

    color: bisque;

    background-color: rgb(184, 93, 18);

    padding: 3px;

}

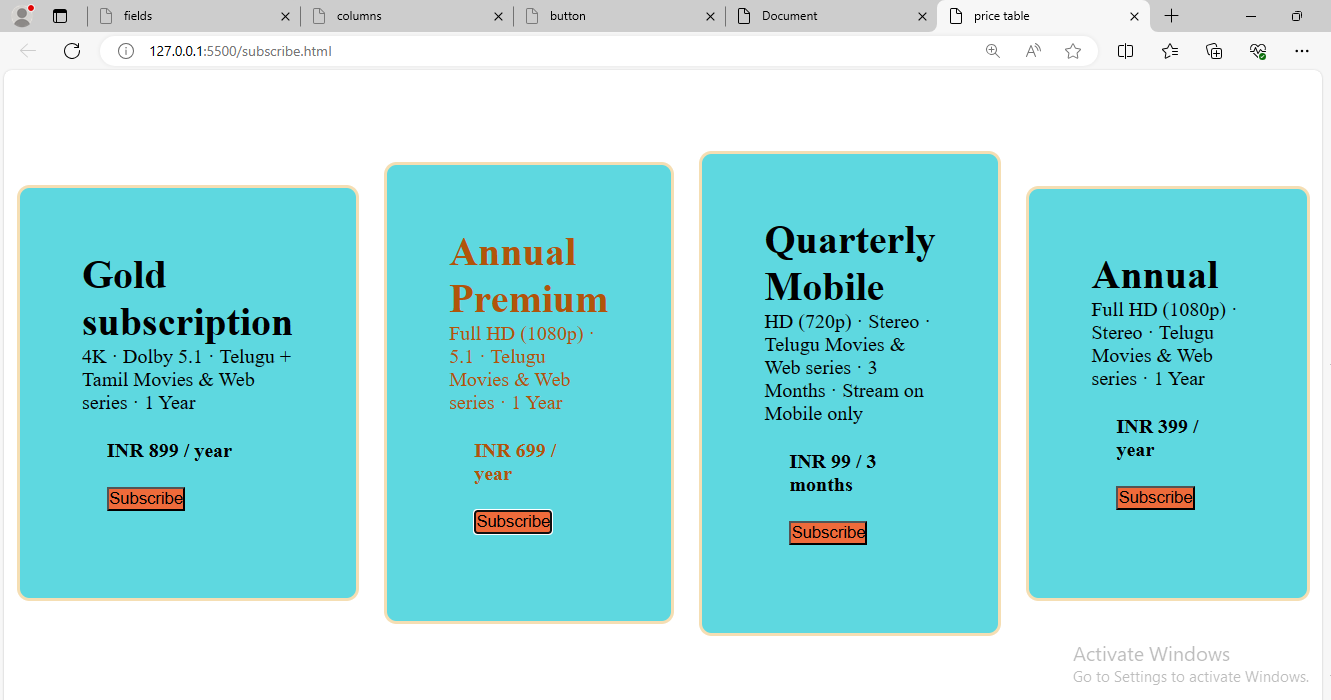
.amount footer{

    margin: 20px;

    font-weight: bolder;

}

**Output:**

****

1. Make a sticky navigation bar that stays at the top of the page when scrolling.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>overflow</title>

    <link rel="stylesheet" href="scrolling.css">

</head>

<body>

    <div class="overflow">

            <nav>

                <a href="">Home</a>

                <a href="">About</a>

                <a href="">Contact Us</a>

            </nav>

            <div>

                <h1>Html,CSS,Javascript</h1>

                <p>Data comes to orange is formatted quickly to the desired pattern, and moving the widgets can be easily transferred where needed. Orange is quite interesting to users. Orange allows its users to make smarter decisions in a short time by rapidly comparing and analyzing the data.It is a good open-source data visualization as well as evaluation that concerns beginners and professionals. Data mining can be performed via visual programming or Python scripting. Many analyses are feasible through its visual programming interface(drag and drop connected with widgets)and many visual tools tend to be supported such as bar charts, scatterplots, trees, dendrograms, and heat maps. A substantial amount of widgets(more than 100) tend to be supported.

                    The instrument has machine learning components, add-ons for bioinformatics and text mining, and it is packed with features for data analytics. This is also used as a python library.Python scripts can keep running in a terminal window, an integrated environment like PyCharmand PythonWin, pr shells like iPython. Orange comprises of canvas interface onto which the user places widgets and creates a data analysis workflow. The widget proposes fundamental operations, For example, reading the data, showing a data table, selecting features, training predictors, comparing learning algorithms, visualizing data elements, etc. Orange operates on Windows, Mac OS X, and a variety of Linux operating systems. Orange comes with multiple regression and classification algorithms.

                    Orange can read documents in native and other data formats. Orange is dedicated to machine learning techniques for classification or supervised data mining. There are two types of objects used in classification: learner and classifiers. Learners consider class-leveled data and return a classifier. Regression methods are very similar to classification in Orange, and both are designed for supervised data mining and require class-level data. The learning of ensembles combines the predictions of individual models for precision gain. The model can either come from different training data or use different learners on the same sets of data.

                    Learners can also be diversified by altering their parameter sets. In orange, ensembles are simply wrappers around learners. They act like any other learner. Based on the data, they return models that can predict the results of any data instance.Rapid Miner is one of the most popular predictive analysis systems created by the company with the same name as the Rapid Miner. It is written in JAVA programming language. It offers an integrated environment for text mining, deep learning, machine learning, and predictive analysis

                    The instrument can be used for a wide range of applications, including company applications, commercial applications, research, education, training, application development, machine learning.

                    Rapid Miner provides the server on-site as well as in public or private cloud infrastructure. It has a client/server model as its base. A rapid miner comes with template-based frameworks that enable fast delivery with few errors(which are commonly expected in the manual coding writing process)

                      Orange is a perfect machine learning and data mining software suite. It supports the visualization and is a software-based on components written in Python computing language and developed at the bioinformatics laboratory at the faculty of computer and information science, Ljubljana University, Slovenia.

As it is a software-based on components, the components of Orange are called "widgets." These widgets range from preprocessing and data visualization to the assessment of algorithms and predictive modeling In other words, we can say that Data Mining is the process of investigating hidden patterns of information to various perspectives for categorization into useful data, which is collected and assembled in particular areas such as data warehouses, efficient analysis, data mining algorithm, helping decision making and other data requirement to eventually cost-cutting and generating revenue.

Data mining is the act of automatically searching for large stores of information to find trends and patterns that go beyond simple analysis procedures. Data mining utilizes complex mathematical algorithms for data segments and evaluates the probability of future events. Data Mining is also called Knowledge Discovery of Data (KDD).

Data Mining is a process used by organizations to extract specific data from huge databases to solve business problems. It primarily turns raw data into useful information.

Data Mining is similar to Data Science carried out by a person, in a specific situation, on a particular data set, with an objective. This process includes various types of services such as text mining, web mining, audio and video mining, pictorial data mining, and social media mining. It is done through software that is simple or highly specific. By outsourcing data mining, all the work can be done faster with low operation costs. Specialized firms can also use new technologies to collect data that is impossible to locate manually. There are tonnes of information available on various platforms, but very little knowledge is accessible. The biggest challenge is to analyze the data to extract important information that can be used to solve a problem or for company development. There are many powerful instruments and techniques available to mine data and find better insight from it.

A relational database is a collection of multiple data sets formally organized by tables, records, and columns from which data can be accessed in various ways without having to recognize the database tables. Tables convey and share information, which facilitates data searchability, reporting, and organization.

Data warehouses:

A Data Warehouse is the technology that collects the data from various sources within the organization to provide meaningful business insights. The huge amount of data comes from multiple places such as Marketing and Finance. The extracted data is utilized for analytical purposes and helps in decision- making for a business organization. The data warehouse is designed for the analysis of data rather than transaction processing.

Data Repositories:

The Data Repository generally refers to a destination for data storage. However, many IT professionals utilize the term more clearly to refer to a specific kind of setup within an IT structure. For example, a group of databases, where an organization has kept various kinds of information.

Object-Relational Database:

A combination of an object-oriented database model and relational database model is called an object-relational model. It supports Classes, Objects, Inheritance, etc.

Data Mining in Healthcare:

Data mining in healthcare has excellent potential to improve the health system. It uses data and analytics for better insights and to identify best practices that will enhance health care services and reduce costs. Analysts use data mining approaches such as Machine learning, Multi-dimensional database, Data visualization, Soft computing, and statistics. Data Mining can be used to forecast patients in each category. The procedures ensure that the patients get intensive care at the right place and at the right time. Data mining also enables healthcare insurers to recognize fraud and abuse.

Data Mining in Market Basket Analysis:

Market basket analysis is a modeling method based on a hypothesis. If you buy a specific group of products, then you are more likely to buy another group of products. This technique may enable the retailer to understand the purchase behavior of a buyer. This data may assist the retailer in understanding the requirements of the buyer and altering the store's layout accordingly. Using a different analytical comparison of results between various stores, between customers in different demographic groups can be done.

Data mining in Education:

Education data mining is a newly emerging field, concerned with developing techniques that explore knowledge from the data generated from educational Environments. EDM objectives are recognized as affirming student's future learning behavior, studying the impact of educational support, and promoting learning science. An organization can use data mining to make precise decisions and also to predict the results of the student. With the results, the institution can concentrate on what to teach and how to teach.

</p>

            </div>

    </div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.overflow  nav{

    display:inline-block;

    position: fixed;

    overflow: hidden;

    background-color: rgb(161, 184, 34);

    padding: 30px;

    top: 0px;

    width: 100%;

    justify-content: space-between;

}

.overflow nav a{

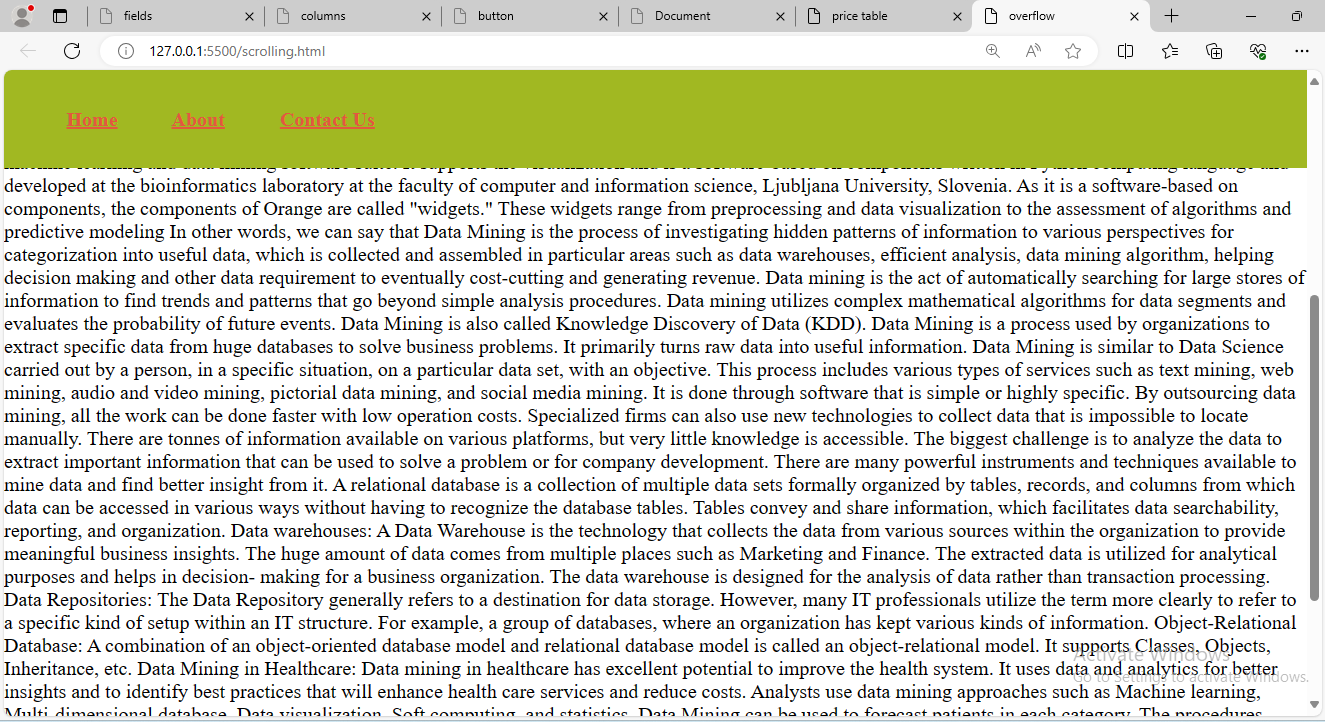
    margin: 20px;

    color:rgb(226, 88, 64);

    font-weight: bolder;

}

**Output:**

****

1. Implement a hover effect on images that enlarges them slightly.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>image</title>

    <link rel="stylesheet" href="image.css">

</head>

<body>

    <div>

        <img src="download.png" alt="google">

    </div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

div{

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

div img{

    width: 50%;

    height: 150px;

    border: 5px solid rgb(240, 171, 171);

    border-radius: 50px;

}

div img:hover{

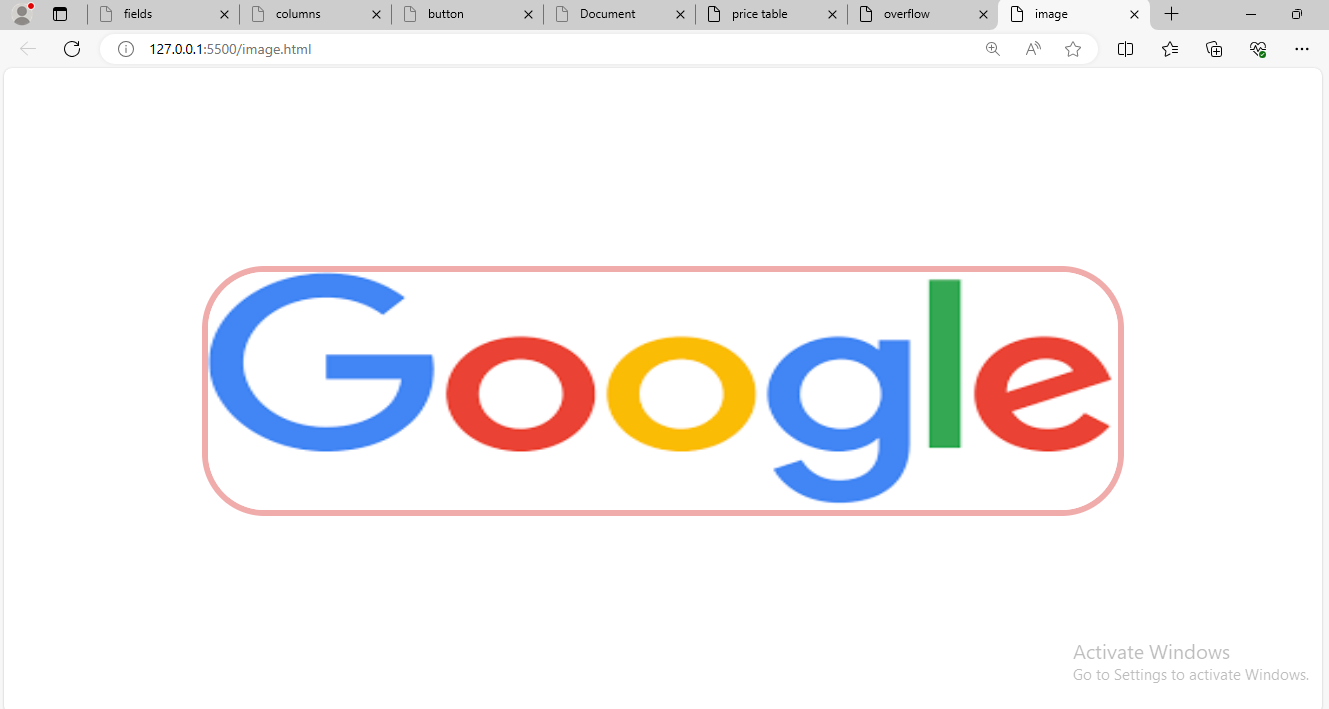
    width: 70%;

    height:200px ;

    cursor: pointer;

}

**Output:**

****

1. Create a login form with HTML and style it using CSS.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>login form</title>

    <link rel="stylesheet" href="login.css">

</head>

<body>

    <div class="total-page">

        <form>

            <h1>Login Form</h1>

            <div class="content">

                <label for="email">Email:</label>

                <input type="text" placeholder="enter here" id="email" name="email" required>

            </div>

            <div class="content">

                <label for="password">Password:</label>

                <input type="text" placeholder="enter here" id="password" name="password" required>

            </div>

            <button>login</button>

        </form>

    </div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.total-page{

    display:flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

.total-page form{

    border: 3px solid greenyellow;

    border-radius: 5px;

    padding: 50px;

    background-color: aqua;

}

.total-page form .content input{

    border: 2px solid whitesmoke;

    border-radius: 5px;

    padding: 5px;

    margin: 5px;

    width: 100%;

}

.total-page form h1{

    text-align: center;

    font-size:large;

    margin-bottom: 20px;

    font-weight: bolder;

}

.total-page form button{

    width: 100%;

  padding: 3px;

  border-radius: 6px;

  margin: 5px;

  background-color: rgb(253, 113, 246);

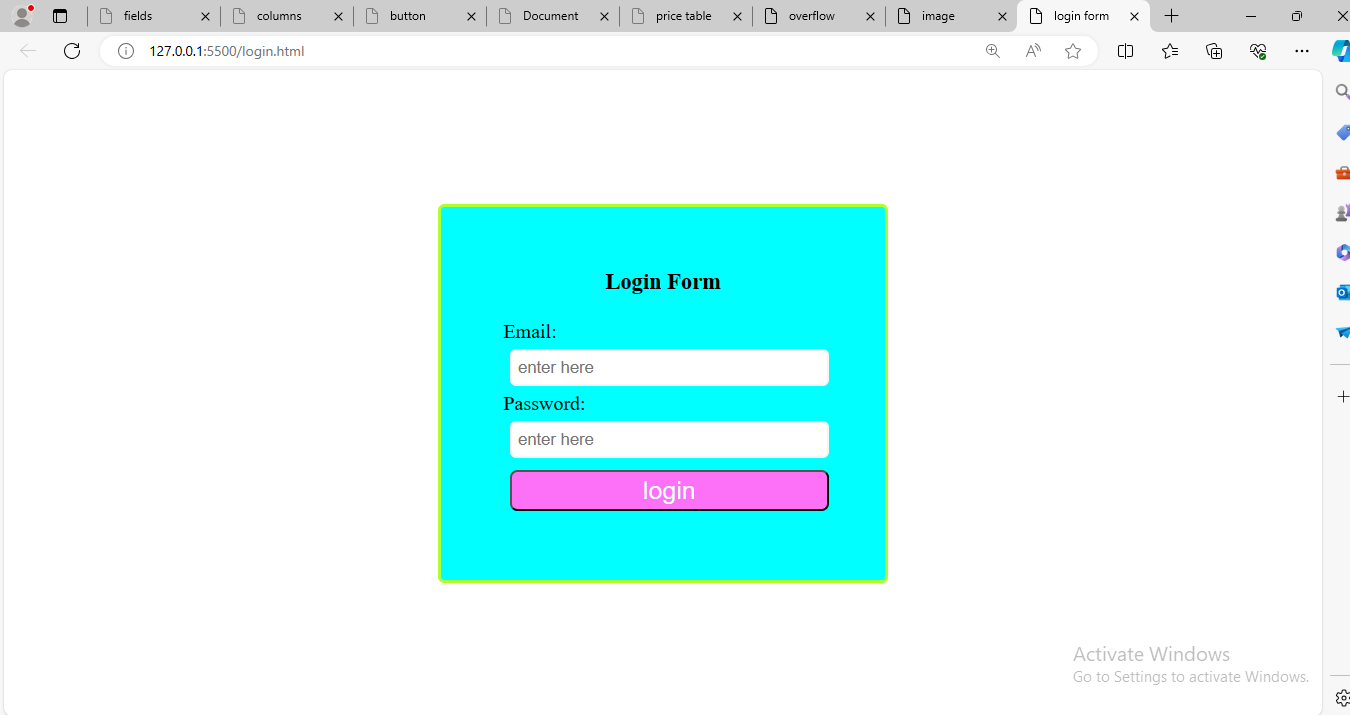
  color: rgb(240, 255, 254);

  font-size: 20px;

  cursor: pointer;

}

**Output:**

****

1. Build a responsive gallery that displays images in a grid format.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Gallery</title>

    <link rel="stylesheet" href="gallery.css">

</head>

<body>

    <div class="total-page">

        <div>

            <h1>image 1</h1>

            <img src="images.jpeg" alt="image">

        </div>

        <div>

            <h1>image 2</h1>

            <img src="images (1).jpeg" alt="image">

        </div>

        <div>

            <h1>image 3</h1>

            <img src="images (2).jpeg" alt="image">

        </div>

        <div>

            <h1>image 4</h1>

            <img src="images (3).jpeg" alt="image">

        </div>

        <div>

            <h1>image 5</h1>

            <img src="images (4).jpeg" alt="image">

        </div>

        <div>

            <h1>image 6</h1>

            <img src="images (5).jpeg" alt="image">

        </div>

    </div>

</body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.total-page{

    display: grid;

  grid-template-columns: repeat(3, 1fr);

  grid-gap: 10px;

}

.total-page div img{

    width: 200px;

  height: 200px;

  margin: 10px;

  border: 2px solid lightgreen;

  border-radius: 20px;

  box-shadow: 2px 2px 15px rgb(147, 114, 114);

}

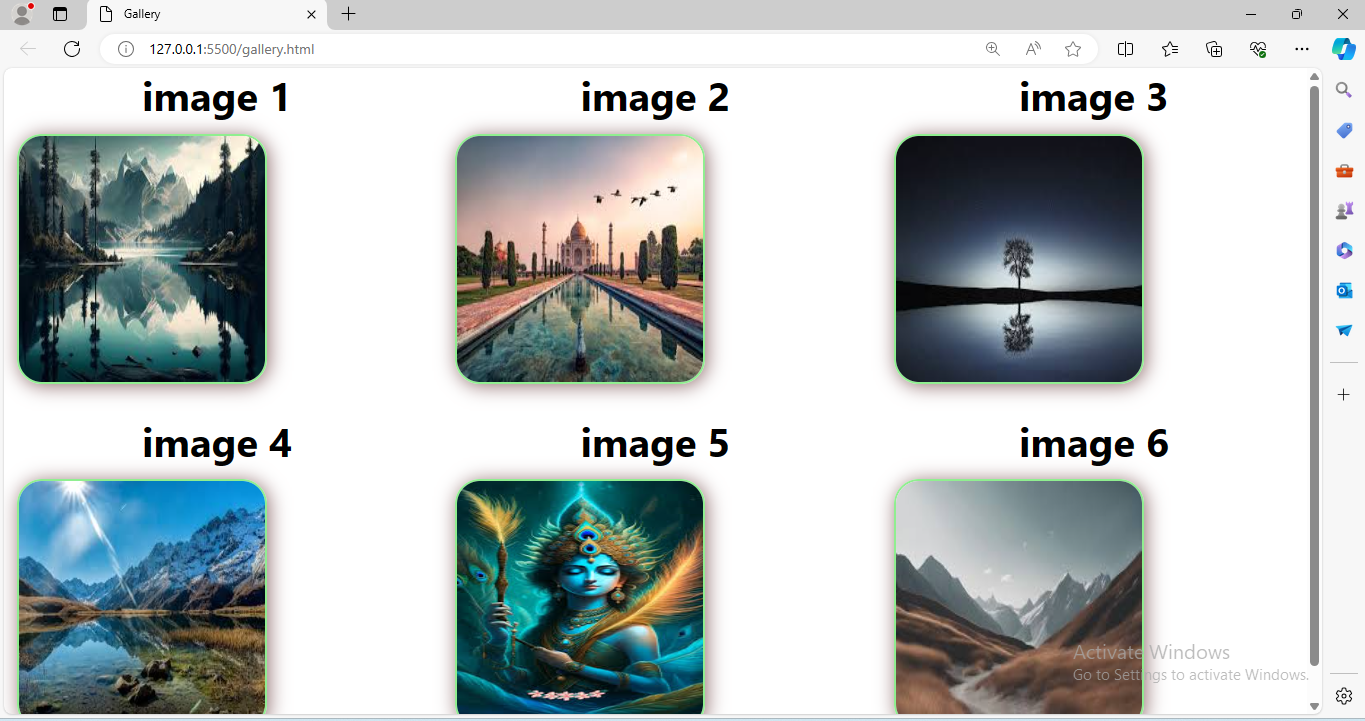
.total-page div h1{

    text-align: center;

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

**Output:**

****

1. Style text to have different fonts, sizes, and colors on the same page.

**Html:**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>font,sizes,color</title>

    <link rel="stylesheet" href="font.css" />

  </head>

  <body>

    <div class="total-page">

      <div class="fonts">

        <h1>Data analytics</h1>

        <p class="intro">converts raw data into actionable insights</p>

        <p class="base">descriptive analytics</p>

        <p class="visit"> predictive analytics</p>

      </div>

    </div>

  </body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

}

.total-page{

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}

.total-page .fonts h1{

    font-family: "Gill Sans", "Gill Sans MT", Calibri, "Trebuchet MS", sans-serif;

  font-size: 30px;

  color: #04a18c;

  font-weight: bold;

}

.total-page .fonts .intro{

    font-family: "Trebuchet MS", "Lucida Sans Unicode", "Lucida Grande",

    "Lucida Sans", Arial, sans-serif;

  font-size: 20px;

  color: rgb(210, 98, 98);

  font-variant: small-caps;

}

.total-page .fonts .base{

    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;

  font-weight: 600;

  color: rgb(209, 17, 17);

  font-size: 25px;

}

.total-page .fonts .visit{

    font-family: Georgia, "Times New Roman", Times, serif;

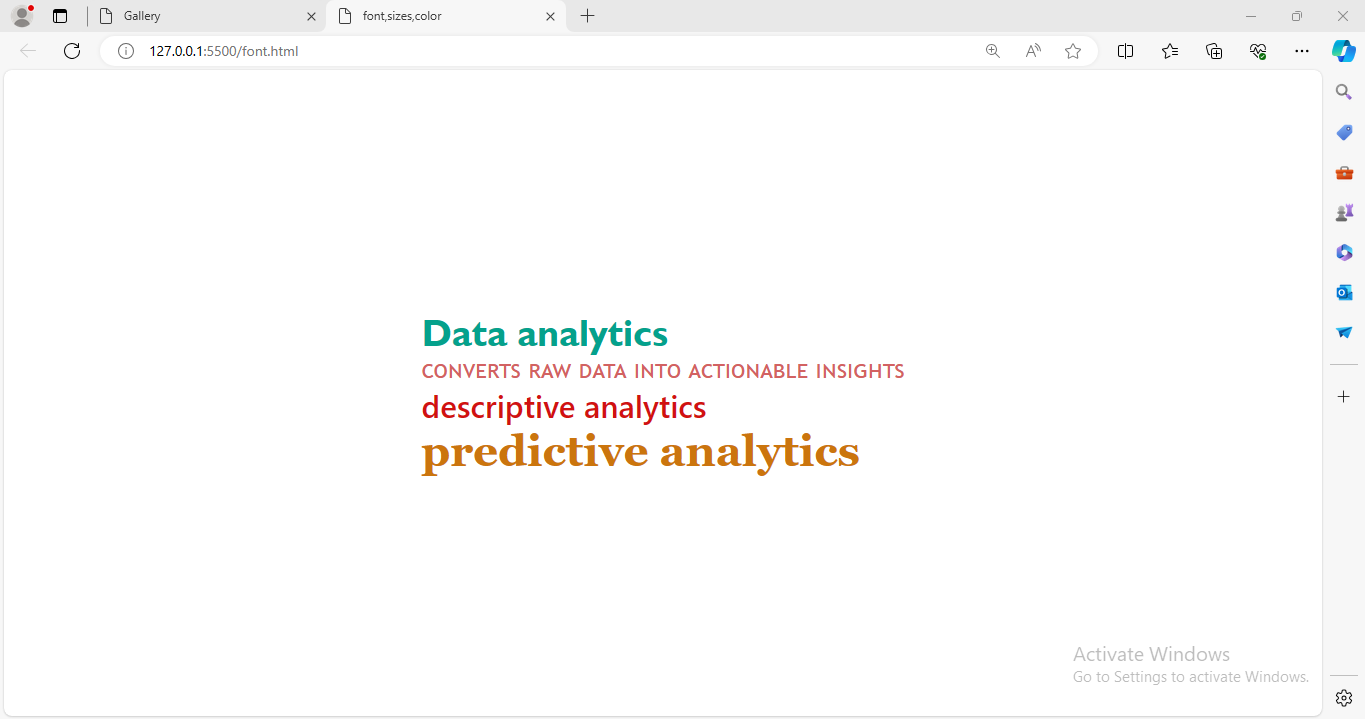
    font-weight: 800;

    color: rgb(203, 117, 13);

    font-size: 35px;

  }

**Output:**

****

1. Design a footer that stays at the bottom of the page, regardless of content height.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>header,footer</title>

    <link rel="stylesheet" href="header.css">

</head>

<body>

    <div class="total-page">

        <section>

          <nav>

            <a href="">Home</a>

            <a href="">About</a>

            <a href="">Contact</a>

          </nav>

          <div class="form">

            <form>

              <div class="fields">

                <label for="username">Username:</label>

                <input

                  type="text"

                  placeholder="Enter here"

                  id="username"

                  name="username"

                />

              </div>

              <div class="fields">

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

                <input

                  type="password"

                  placeholder="Enter here"

                  id="password"

                  name="password"

                />

              </div>

              <button>login</button>

            </form>

          </div>

        </section>

        <footer class="footer">

          <p>@it is a login page</p>

        </footer>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page section nav {

    display: block;

    position: relative;

    background-color: rgb(248, 26, 148);

    padding: 20px;

    top: 0px;

  }

  .total-page section nav a {

    text-decoration: none;

    margin: 20px;

    color: rgb(235, 241, 241);

  }

  .footer {

    position: absolute;

    width: 100%;

    line-height: 50px;

    height: 50px;

    background-color: #023637;

    color: #fff;

    text-align: center;

    bottom: 0;

  }

  .form{

    display: flex;

    justify-content: center;

    align-items: center;

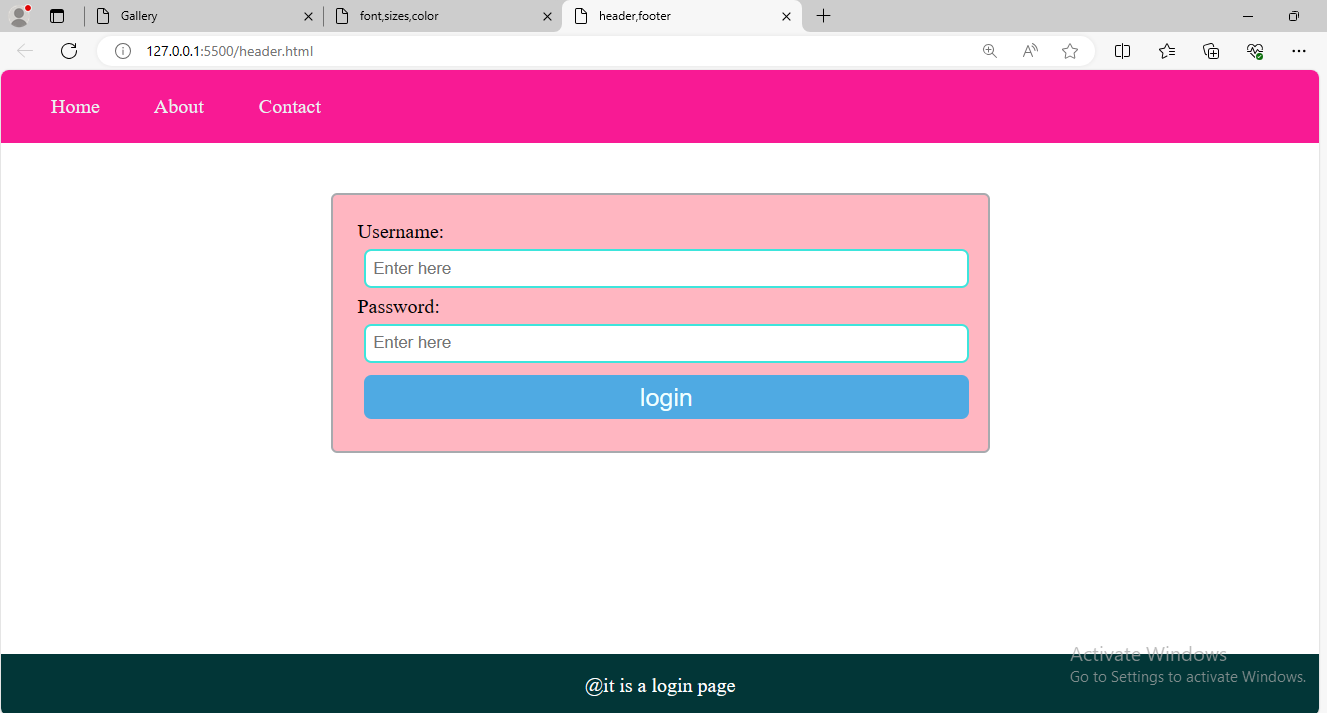
    margin-top: 40px;

  }

  .form form h2 {

    text-a

**output:**

****

1. Make a horizontal scrolling website using CSS overflow property.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>scrolling</title>

    <link rel="stylesheet" href="using.css">

</head>

<body>

    <div class="total-page">

        <div class="sections">

          <ol>

            <li>vegetables</li>

            <li>non-veg</li>

            <li>leafy-vegetables</li>

            <li>Milk</li>

          </ol>

        </div>

        <div class="sections">

          <ol>

            <li>cucumber</li>

            <li>chicken</li>

            <li>kale</li>

            <li>cow milk</li>

          </ol>

        </div>

        <div class="sections">

          <ul>

            <li>vitamin b1</li>

            <li>vitamin b3</li>

            <li>vitamin A</li>

            <li>vitamin E</li>

          </ul>

        </div>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    width: 100%;

    overflow-x: auto;

    white-space: nowrap;

    margin-top: 70px;

  }

  .sections {

    display: inline-block;

    width: 50%;

    padding: 20px;

    background-color: #d612d6;

    box-sizing: border-box;

  }

  .total-page ol {

    list-style-type: none;

  }

  .total-page ol li {

    display: flex;

    justify-content: center;

    align-items: center;

    font-weight: bold;

    color: rgb(248, 252, 251);

  }

  .total-page ul {

    list-style-type: none;

  }

  .total-page ul li {

    display: flex;

    justify-content: center;

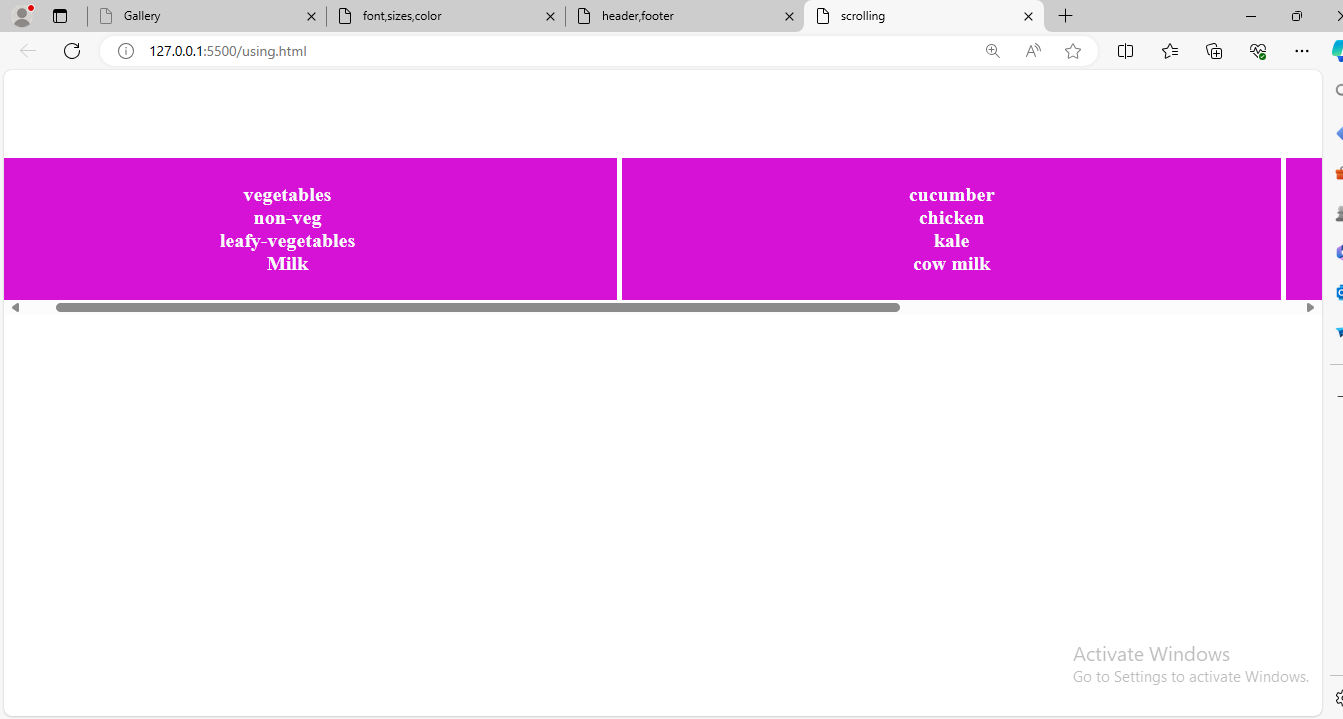
    align-items: center;

    font-weight: bold;

    color: rgb(218, 220, 219);

  }

**Output:**

****

1. Create a progress bar using HTML and CSS.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>skills</title>

    <link rel="stylesheet" href="bar.css">

</head>

<body>

    <div class="total-page">

        <div class="skill">

          <h5>HTML</h5>

          <progress id="file" value="95" max="100">95%</progress>

        </div>

        <div class="skill">

          <h5>CSS</h5>

          <progress id="file" value="80" max="100">80%</progress>

        </div>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    gap: 30px;

    color: rgb(46, 114, 160);

  }

  progress {

    appearance: none;

    width: 150px;

    height: 20px;

  }

  progress::-webkit-progress-bar {

    background-color: #e61e1e;

    border-radius: 20px;

  }

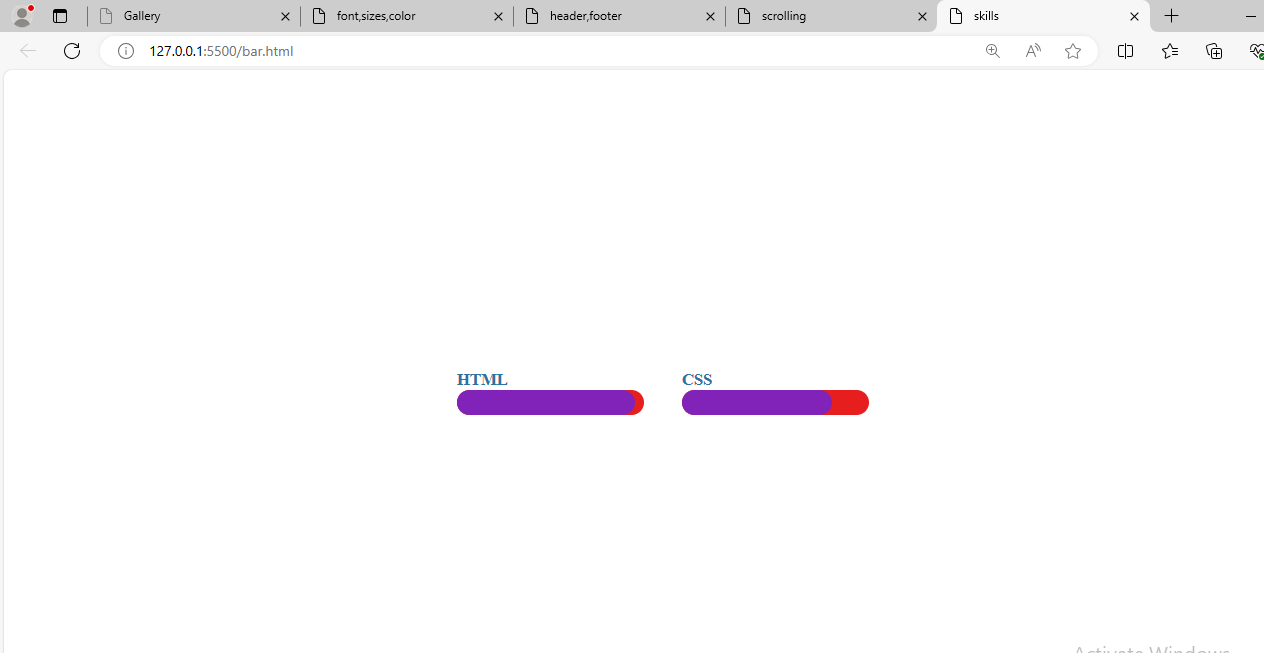
  progress::-webkit-progress-value {

    background-color: #8123b8;

    border-radius: 20px;

  }

**Output:**

****

1. Implement a CSS animation on an element, such as a spinning effect.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>spinning</title>

    <link rel="stylesheet" href="element.css">

</head>

<body>

    <div class="total-page">

        <div class="content">

          <h1>CSS Assignment</h1>

        </div>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

  }

  .total-page .content {

    background-color: rgb(158, 12, 34);

    width: 200px;

    height: 150px;

    border-radius: 40px;

    animation: spin 5s linear infinite;

  }

  .total-page .content h1 {

    text-align: center;

    margin-top: 70px;

    color: rgb(122, 197, 240);

  }

  @keyframes spin {

    from {

      transform: rotate(90deg);

    }

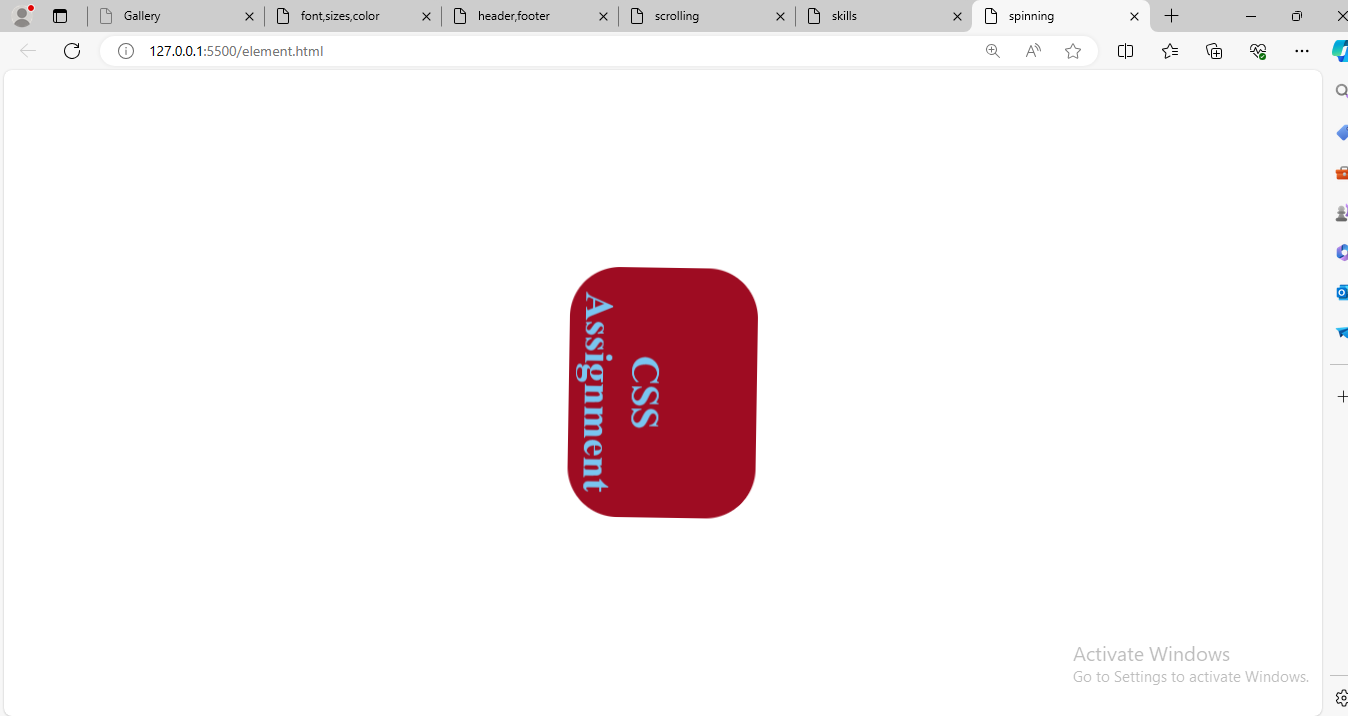
    to {

      transform: rotate(270deg);

    }

  }

**Output:**

****

1. Build a tabbed content section using HTML, CSS, and JavaScript.
2. Design a responsive carousel/slider for showcasing images.

**Html:**

<DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>slider</title>

    <link rel="stylesheet" href="att.css" />

  </head>

  <body>

    <div class="slider">

      <figure>

        <img src="images.jpeg" alt="image" />

        <img src="images (1).jpeg" alt="image" />

        <img src="images (2).jpeg" alt="image" />

      </figure>

    </div>

  </body>

</html>

**Css:**

\*{

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .slider {

    overflow: hidden;

  }

  .slider figure {

    position: relative;

    width: 500%;

    margin: 0px;

    left: 0px;

    animation: 5s slider infinite;

  }

  .slider figure img {

    float: left;

    width: 20%;

    display: inline;

    height: 500px;

    width: 700px;

  }

  @keyframes slider {

    0% {

      left: 0;

    }

    20% {

      left: 0;

    }

    25% {

      left: -100%;

    }

    45% {

      left: -100%;

    }

    50% {

      left: -200%;

    }

    70% {

      left: -200%;

    }

    75% {

      left: -300%;

    }

    95% {

      left: -300%;

    }

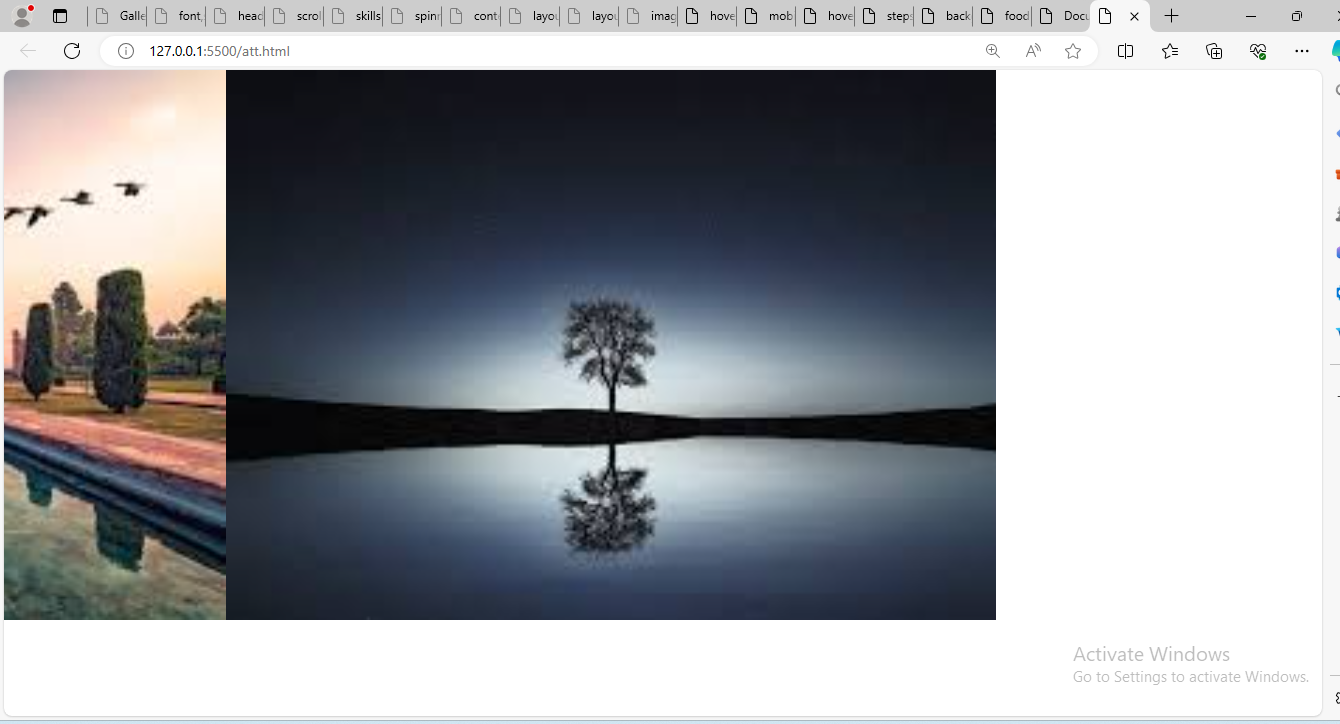
    100% {

      left: -400%;

    }

  }

**Output:**

****

1. Style a list to have custom bullet points and spacing between items.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>content-list</title>

    <link rel="stylesheet" href="content.css">

</head>

<body>

    <div class="total-page">

        <h1>list of the place in karnataka</h1>

        <ul>

          <li>mysore</li>

          <li>coorg</li>

          <li>mangalore</li>

          <li>udupi</li>

        </ul>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    flex-direction: column;

  }

  .total-page ul {

    list-style-type: none;

  }

  .total-page ul li::before {

    content: ".";

    font-size: 30px;

    color: rgb(75, 133, 209);

    position: absolute;

    top: 0;

    left: 0;

  }

  .total-page ul li {

    position: relative;

    padding-left: 25px;

    padding-top: 16px;

  }

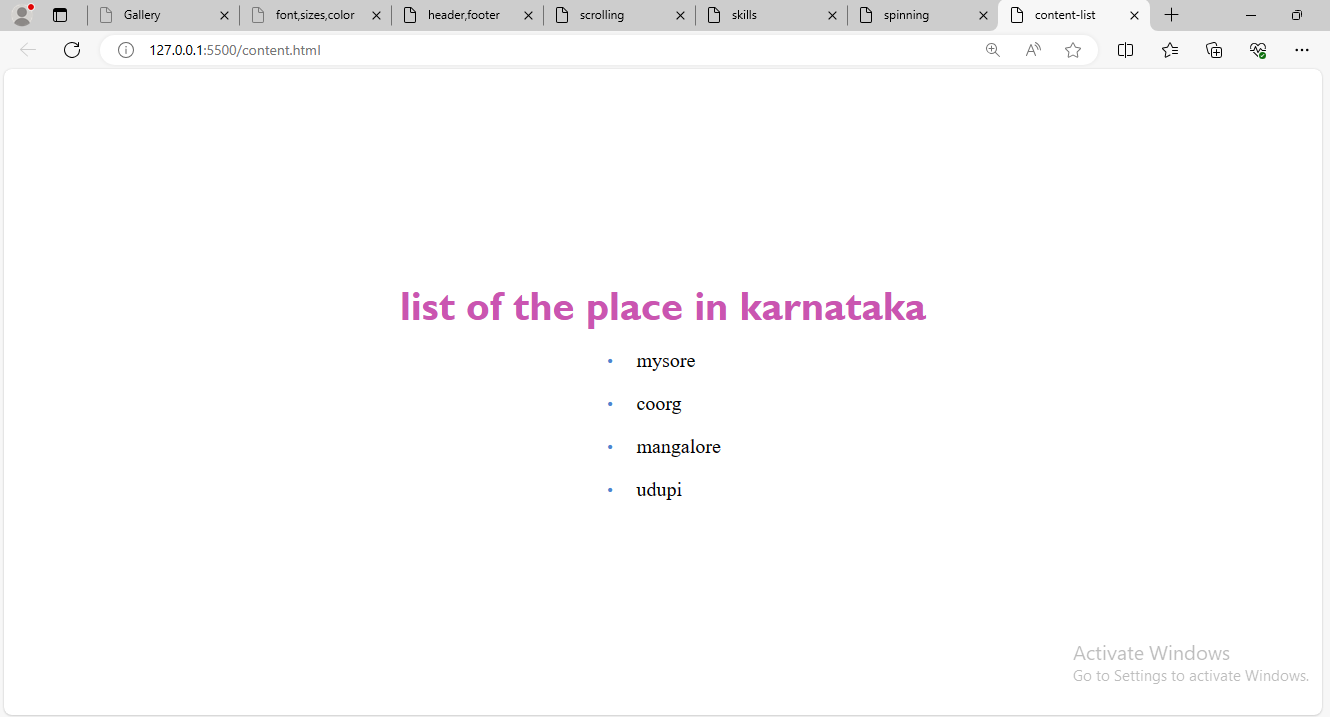
  .total-page h1 {

    color: rgb(202, 85, 177);

    font-family: "Gill Sans", "Gill Sans MT", Calibri, "Trebuchet MS", sans-serif;

  }

**Output:**

****

1. Make a website layout that uses a sticky sidebar.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>layout</title>

    <link rel="stylesheet" href="layout.css">

</head>

<body>

    <div class="total-page">

        <div class="items">

          <a href="">home</a>

          <a href="">about</a>

          <a href="">Contact</a>

        </div>

        <div class="content">

            <p>

Data mining applications

Related solutions

Resources

Take the next step

What is data mining?

Data mining, also known as knowledge discovery in data (KDD), is the process of uncovering patterns and other valuable information from large data sets.

Given the evolution of data warehousing technology and the growth of big data, adoption of data mining techniques has rapidly accelerated over the last couple of decades, assisting companies by transforming their raw data into useful knowledge. However, despite the fact that that technology continuously evolves to handle data at a large scale, leaders still face challenges with scalability and automation.

Data mining has improved organizational decision-making through insightful data analyses. The data mining techniques that underpin these analyses can be divided into two main purposes; they can either describe the target dataset or they can predict outcomes through the use of machine learning algorithms. These methods are used to organize and filter data, surfacing the most interesting information, from fraud detection to user behaviors, bottlenecks and even security breaches.

When combined with data analytics and visualization tools, like Apache Spark, delving into the world of data mining has never been easier and extracting relevant insights has never been faster. Advances within artificial intelligence only continue to expedite adoption across industries.

Guide

A data leader's guide

Learn how to leverage the right databases for applications, analytics and generative AI.

Related content

Register for the ebook on generative AI

What is data mining?

Data mining, also known as knowledge discovery in data (KDD), is the process of uncovering patterns and other valuable information from large data sets. Given the evolution of data warehousing technology and the growth of big data, adoption of data mining techniques has rapidly accelerated over the last couple of decades, assisting companies by transforming their raw data into useful knowledge. However, despite the fact that that technology continuously evolves to handle data at a large scale, leaders still face challenges with scalability and automation.

Data mining has improved organizational decision-making through insightful data analyses. The data mining techniques that underpin these analyses can be divided into two main purposes; they can either describe the target dataset or they can predict outcomes through the use of machine learning algorithms. These methods are used to organize and filter data, surfacing the most interesting information, from fraud detection to user behaviors, bottlenecks and even security breaches.

When combined with data analytics and visualization tools, like Apache Spark, delving into the world of data mining has never been easier and extracting relevant insights has never been faster. Advances within artificial intelligence only continue to expedite adoption across industries.

Now available: watsonx.data

Scale AI workloads for all your data anywhere.

Try watsonx.data

Data mining process

The data mining process involves a number of steps from data collection to visualization to extract valuable information from large data sets. As mentioned above, data mining techniques are used to generate descriptions and predictions about a target data set. Data scientists describe data through their observations of patterns, associations and correlations. They also classify and cluster data through classification and regression methods, and identify outliers for use cases, like spam detection.

Data mining usually consists of four main steps: setting objectives, data gathering and preparation, applying data mining algorithms and evaluating results.

1. Set the business objectives: This can be the hardest part of the data mining process, and many organizations spend too little time on this important step. Data scientists and business stakeholders need to work together to define the business problem, which helps inform the data questions and parameters for a given project. Analysts may also need to do additional research to understand the business context appropriately.

2. Data preparation: Once the scope of the problem is defined, it is easier for data scientists to identify which set of data will help answer the pertinent questions to the business. Once they collect the relevant data, it will be cleaned, removing any noise, such as duplicates, missing values and outliers. Depending on the dataset, an additional step may be taken to reduce the number of dimensions as too many features can slow down any subsequent computation. Data scientists will look to retain the most important predictors to ensure optimal accuracy within any models.

3. Model building and pattern mining: Depending on the type of analysis, data scientists may investigate any interesting data relationships, such as sequential patterns, association rules or correlations. While high-frequency patterns have broader applications, sometimes the deviations in the data can be more interesting, highlighting areas of potential fraud.

Deep learning algorithms may also be applied to classify or cluster a data set depending on the available data. If the input data is labelled (i.e. supervised learning), a classification model may be used to categorize data, or alternatively, a regression may be applied to predict the likelihood of a particular assignment. If the dataset isn’t labelled (i.e. unsupervised learning), the individual data points in the training set are compared with one another to discover underlying similarities, clustering them based on those characteristics.

4. Evaluation of results and implementation of knowledge: Once the data is aggregated, the results need to be evaluated and interpreted. When finalizing results, they should be valid, novel, useful and understandable. When this criteria is met, organizations can use this knowledge to implement new strategies, achieving their intended objectives.

Data mining techniques

Data mining works by using various algorithms

            </p>

          </div>

    </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

  }

  .total-page .items {

    flex: 0 0 20%;

    padding: 20px;

    position: sticky;

    top: 0;

    background-color: rgb(40, 192, 154);

    height: 100vh;

    overflow: hidden;

  }

  .total-page .content {

    flex: 0 0 80%;

  }

  .total-page .items a {

    text-decoration: none;

    display: flex;

    flex-direction: column;

    position: relative;

    left: 50px;

    top: 70px;

    padding-top: 40px;

  }

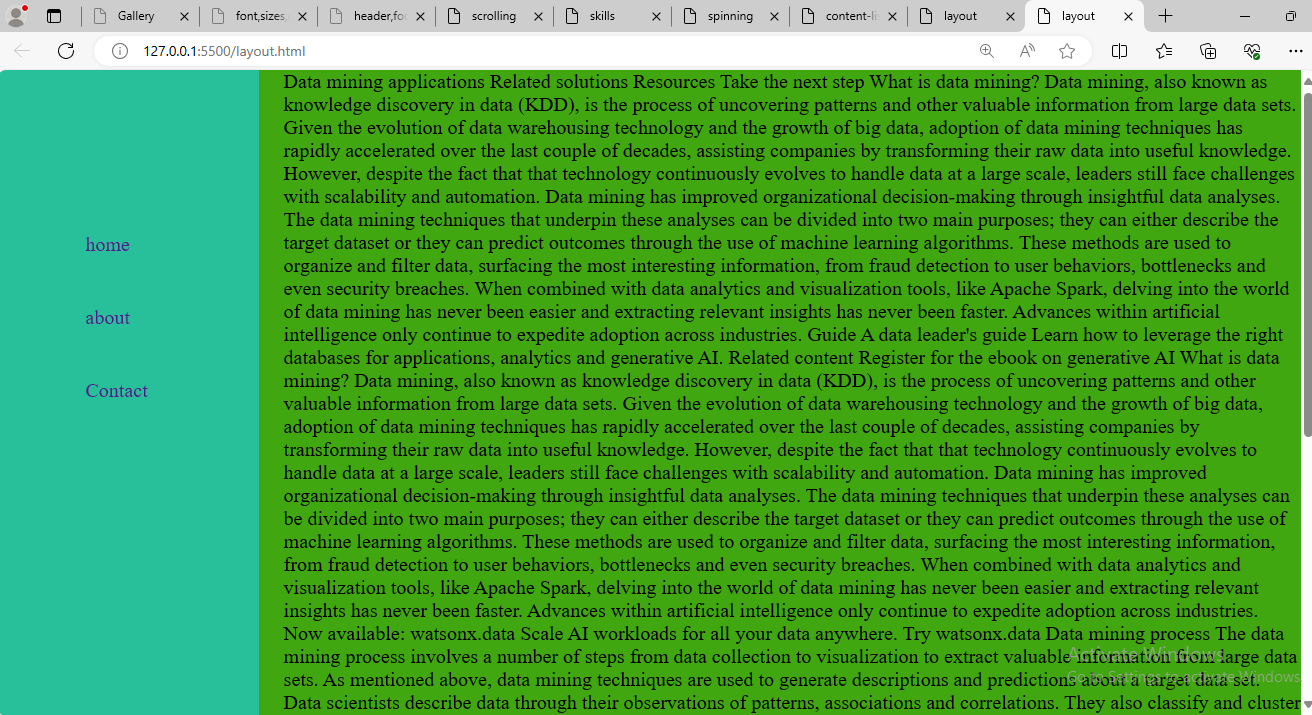
  .total-page .content {

    background-color: rgb(65, 167, 17);

    padding-left: 20px;

  }

**Output:**

****

1. Create a simple tooltip using HTML and CSS.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Document</title>

    <link rel="stylesheet" href="dot.css">

</head>

<body>

    <span>tooltip

     <span class="tooltip">this is a tooltip</span></span>

</body>

</html>

**Css:**

\*{

  margin: 0px;

  padding: 0px;

  box-sizing: border-box;

}

.tooltip {

    position: absolute;

    border: none;

    background-color: #fa3c58;

    width: 150px;

    min-height: 20px;

    color: #ffff;

    padding: 0.3rem;

    align-items: center;

    border: 0.5rem;

    font-size: 1rem;

    bottom: 2rem;

    right: -3rem;

    visibility: hidden;

    cursor: help;

    border-radius: 10px;

  }

  .tooltip::after {

    content: "";

    position: absolute;

    border-left: 10px solid transparent;

    border-bottom: 10px solid transparent;

    border-right: 10px solid transparent;

    border-top: 10px solid #fa3c58;

    top: 33.5px;

    left: 78px;

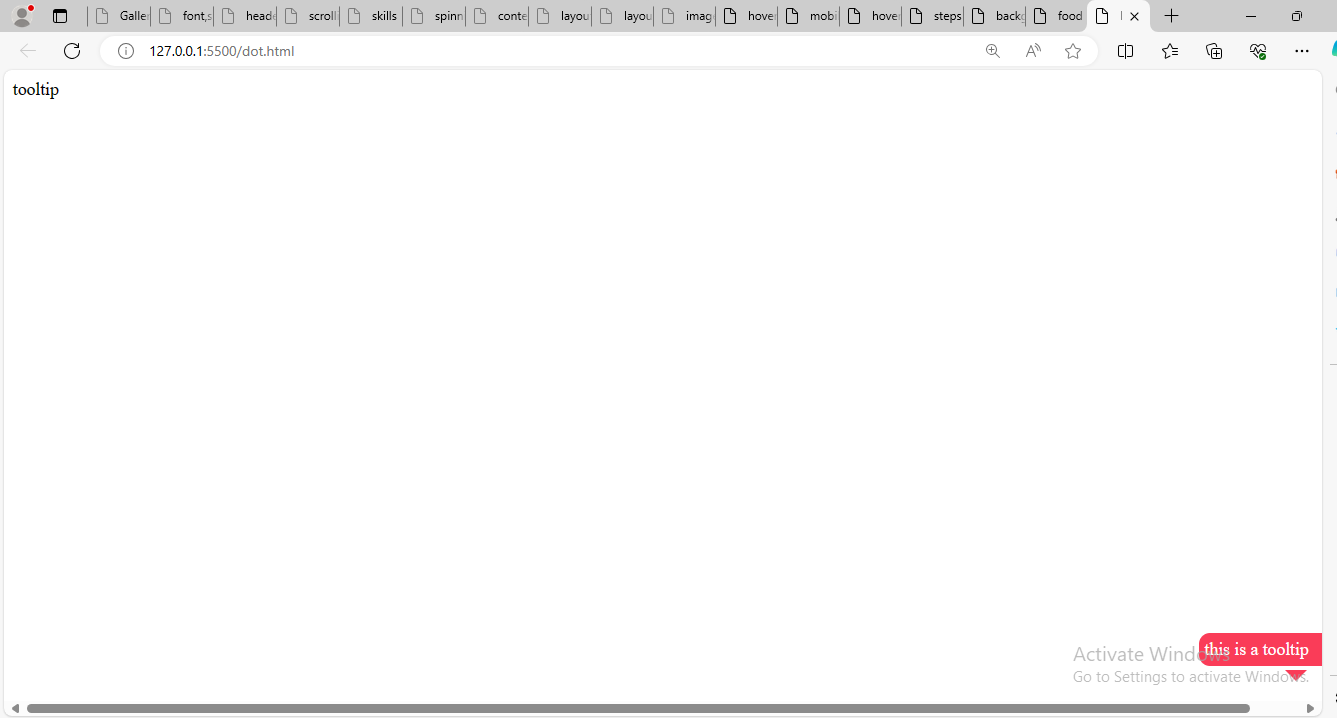
  }

  span:hover .tooltip {

    visibility: visible;

  }

**Output:**

****

1. Design a card layout for displaying content with an image, title, and description.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>images</title>

    <link rel="stylesheet" href="title.css">

</head>

<body>

    <div class="total-page">

        <div class="cards">

          <img src="images.jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">description area</p>

          </div>

        </div>

        <div class="cards">

          <img src="images (1).jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">description area</p>

          </div>

        </div>

        <div class="cards">

          <img src="images (2).jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">discription area></p>

          </div>

        </div>

        <div class="cards">

          <img src="images (3).jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area">title area</h3>

            <p class="about-card">discription area </p>

          </div>

        </div>

        <div class="cards">

          <img src="images (4).jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">discription area</p>

          </div>

        </div>

        <div class="cards">

          <img src="images (5).jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">discription area</p>

          </div>

        </div>

        <div class="cards">

          <img src="download (1).jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">discription area</p>

          </div>

        </div>

        <div class="cards">

          <img src="download.jpeg" alt="Image" />

          <div class="about-area">

            <h3 class="title-area"> title area</h3>

            <p class="about-card">discription area</p>

          </div>

        </div>

      </div>

</body>

</html>

**Css:**

\*{

margin: 0px;

padding: 0px;

box-sizing: border-box;

}

.total-page {

display: grid;

grid-template-columns: repeat(3, 1fr);

grid-gap: 20px;

justify-content: center;

padding: 20px;

}

.cards {

border-radius: 10px;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

overflow: hidden;

transition: transform 0.3s;

}

.cards img {

width: 100%;

height: 200px;

object-fit: cover;

}

.about-area {

padding: 20px;

}

.title-area {

font-size: 20px;

font-weight: 300;

margin-bottom: 8px;

}

.about-card {

color: #724343;

font-size: 24px;

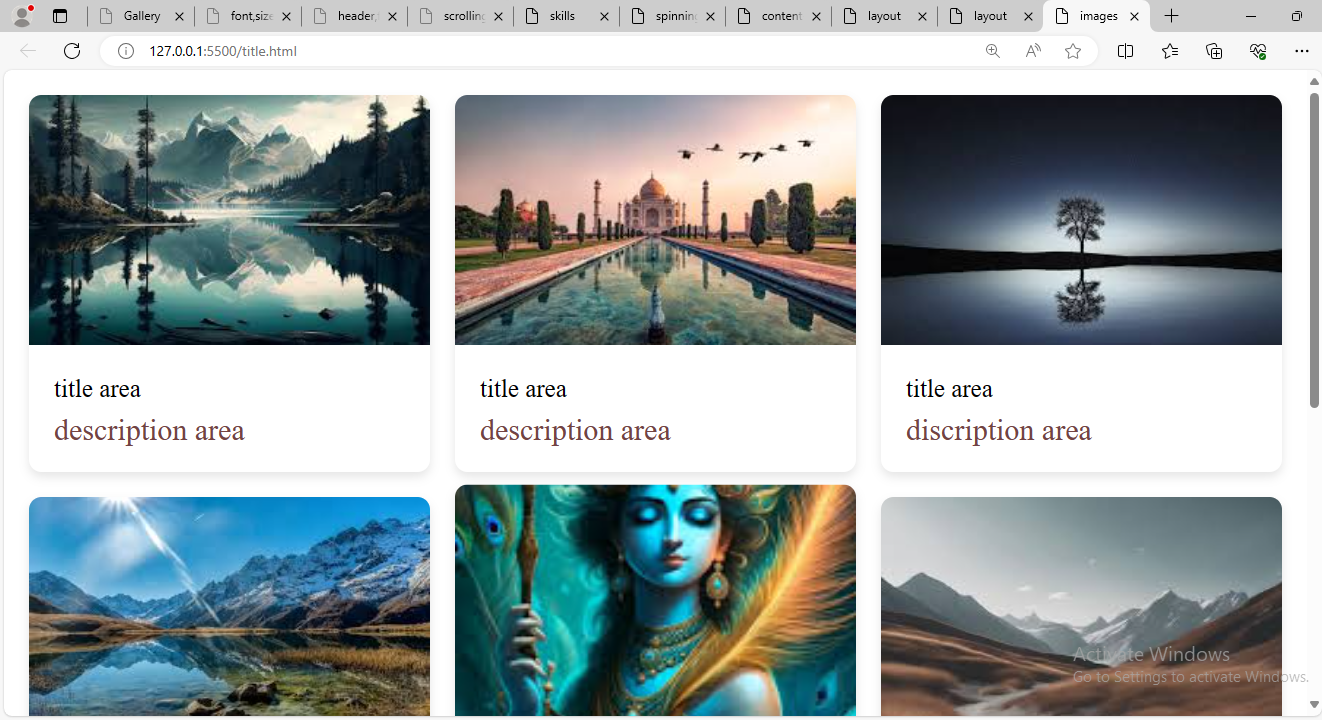
}

.cards:hover {

transform: translateY(-10px);

}

**Output:**

****

1. Implement a CSS transition on a button to change color smoothly.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>hover color</title>

    <link rel="stylesheet" href="color.css">

</head>

<body>

    <div class="total-page">

        <form class="fields">

          <div class="name">

            <label for="name">name</label>

            <input

              type="text"

              id="name"

              name="name"

              placeholder="Enter the name here!"

            />

          </div>

          <input class="button" type="submit" value="submit" />

        </form>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

  }

  .fields {

    border: 2px solid rgb(167, 49, 157);

    background-color: rgb(144, 140, 206);

    padding: 20px;

    width: 50%;

  }

  .name input {

    width: 100%;

    padding: 6px;

  }

  .button {

    width: 100%;

    padding: 6px;

    background-color: rgb(16, 243, 148);

    border: none;

    color: black;

    margin-top: 10px;

  }

  .button:hover {

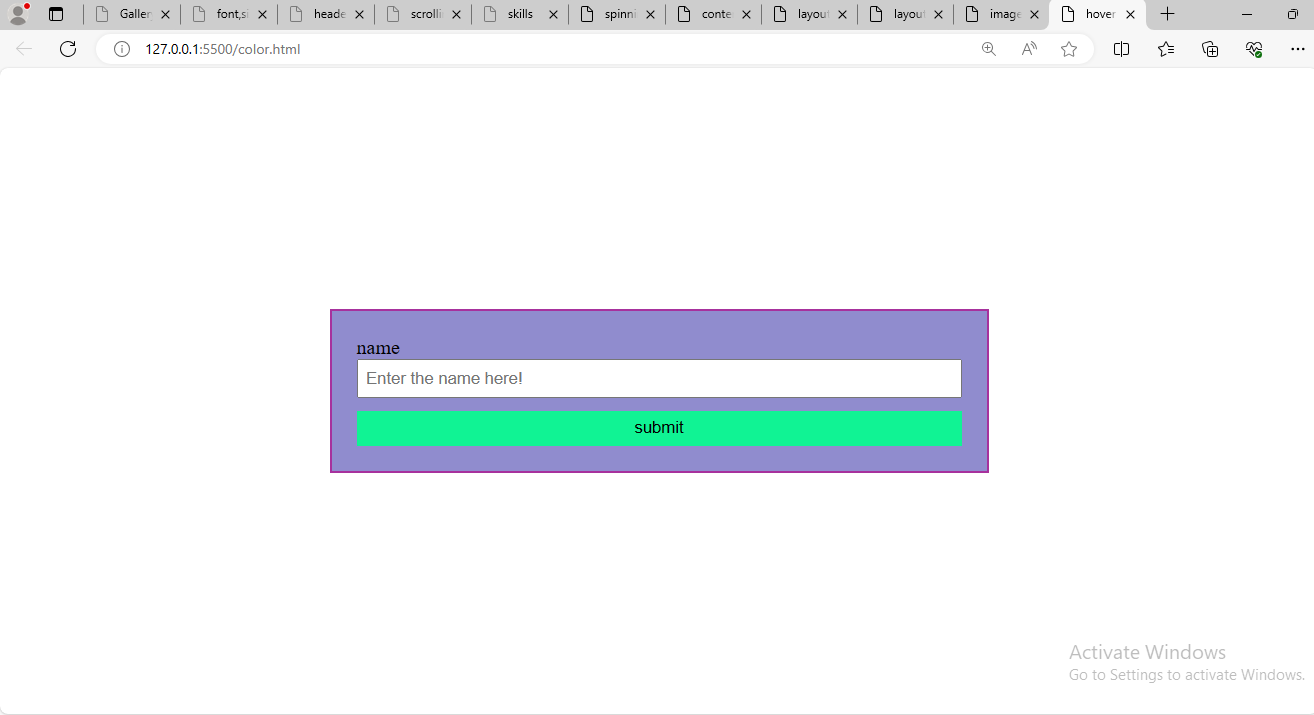
    background-color: rgb(202, 200, 73);

    color: #ffff;

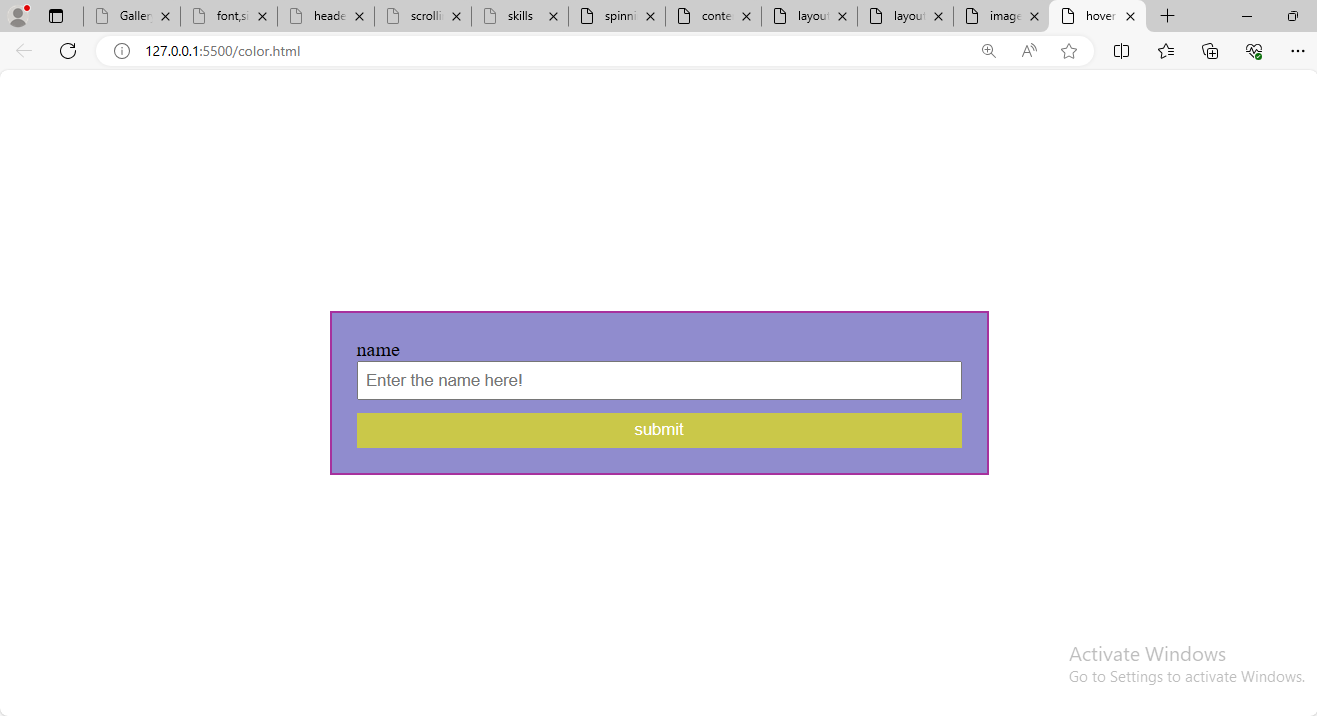
  }

**Output:**

**Without hover:**

****

**With hover:**

****

1. .Build a responsive navbar with a hamburger menu for mobile devices.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>mobile</title>

    <link rel="stylesheet" href="mobile.css">

    <link

    rel="stylesheet"

    href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.2/css/all.min.css"

    integrity="sha512-SnH5WK+bZxgPHs44uWIX+LLJAJ9/2PkPKZ5QiAj6Ta86w+fsb2TkcmfRyVX3pBnMFcV7oQPJkl9QevSCWr3W6A=="

    crossorigin="anonymous"

    referrerpolicy="no-referrer"

  />

</head>

<body>

  <div class="total-page">

    <section>

      <nav>

        <a href="">Home</a>

        <a href="">About us</a>

        <a href="">Contact us</a>

        <i class="fa fa-bars"></i>

      </nav>

    <div>

        <h1>mobile devices</h1>

        <p>Mobile (/moʊˈbiːl/ moh-BEEL, French: [mɔbil] ⓘ) is a city and the county seat of Mobile County, Alabama, United States. The population was 187,041 at the 2020 census.[8][9] After a successful vote to annex areas west of the city limits in July 2023, Mobile's population increased to 204,689 residents, making it the second-most populous city in Alabama, after Huntsville.[11] Mobile is the principal municipality of the Mobile metropolitan area, a region of 430,197 residents composed of Mobile and Washington counties; it is the third-largest metropolitan area in the state.[12]

            Alabama's only saltwater port, Mobile is located on the Mobile River at the head of Mobile Bay on the north-central Gulf Coast.[13] The Port of Mobile has always played a key role in the economic health of the city, beginning with the settlement as an important trading center between the French colonists and Native Americans, down to its current role as the 12th-largest port in the United States.[14][15]

            Mobile was founded in 1702 by the French as the first capital of Louisiana. During its first 100 years, Mobile was a colony of France, then Great Britain, and lastly Spain. Mobile became a part of the United States in 1813, with the annexation by President James Madison of West Florida from Spain.[16] During the American Civil War, the city surrendered to Federal forces on April 12, 1865,[17] after Union victories at two forts protecting the city. This, along with the news of Johnston's surrender negotiations with Sherman, led General Richard Taylor to seek a meeting with his Union counterpart, Maj. Gen. Edward R. S. Canby. The two generals met several miles north of Mobile on May 2. After agreeing to a 48-hour truce, the generals enjoyed an al fresco luncheon of food, drink, and lively music. Canby offered Taylor the same terms agreed upon between Lee and Grant at Appomattox. Taylor accepted the terms and surrendered his command on May 4 at Citronelle, Alabama.[18]

            Considered one of the Gulf Coast's cultural centers, Mobile has several art museums, a symphony orchestra, professional opera, professional ballet company, and a large concentration of historic architecture.[19][20] Mobile is known for having the oldest organized Carnival or Mardi Gras celebrations in the United States. Alabama's French Creole population celebrated this festival from the first decade of the 18th century. Beginning in 1830, Mobile was host to the first formally organized Carnival mystic society to celebrate with a parade in the United States. (In New Orleans, such a group is called a krewe.)[21]

            Etymology

            Further information: Alabama Creole people

            The city gained its name from the Mobile tribe that the French colonists encountered living in the area of Mobile Bay.[22] Although it is debated by Alabama historians, they may have been descendants of the Native American tribe whose small fortress town, Mabila, was used to conceal several thousand native warriors before an attack in 1540 on the expedition of Spanish explorer Hernando de Soto.[23] About seven years after the founding of the French Mobile settlement, the Mobile tribe, along with the Tohomé, gained permission from the colonists to settle near the fort.[24][25] The European settlement of Mobile began with French colonists, who in 1702 constructed Fort Louis de la Louisiane, at Twenty-seven Mile Bluff on the Mobile River, as the first capital of the French colony of La Louisiane. It was founded by French Canadian brothers Pierre Le Moyne d'Iberville and Jean-Baptiste Le Moyne, Sieur de Bienville, to establish control over France's claims to La Louisiane. Bienville was appointed as royal governor of French Louisiana in 1701. Mobile's Roman Catholic parish was established on July 20, 1703, by Jean-Baptiste de la Croix de Chevrières de Saint-Vallier, Bishop of Quebec.[26] The parish was the first French Catholic parish established on the Gulf Coast of the United States.[26]

            In 1704, the ship Pélican delivered 23 Frenchwomen to the colony; passengers had contracted yellow fever at a stop in Havana.[27] Though most of the "Pélican girls" recovered, numerous colonists and neighboring Native Americans contracted the disease in turn and many died.[27] This early period was also the occasion of the importation of the first African slaves, transported aboard a French supply ship from the French colony of Saint-Domingue in the Caribbean, where they had first been held.[27] The population of the colony fluctuated over the next few years, growing to 279 persons by 1708, yet shrinking to 178 persons two years later due to disease.[26]

            These additional outbreaks of disease and a series of floods resulted in Bienville ordering in 1711 that the settlement be relocated several miles downriver to its present location at the confluence of the Mobile River and Mobile Bay.[28] A new earth-and-palisade Fort Louis was constructed at the new site during this time.[29] By 1712, when Antoine Crozat was appointed to take over administration of the colony, its population had reached 400 persons.

            The capital of La Louisiane was moved in 1720 to Biloxi,[29] leaving Mobile to serve as a regional military and trading center. In 1723 the construction of a new brick fort with a stone foundation began[29] and it was renamed Fort Condé in honor of Louis Henri, Duke of Bourbon.[30]

            In 1763, the Treaty of Paris was signed, ending the Seven Years' War, which Britain won, defeating France. By this treaty, France ceded its territories east of the Mississippi River to Britain. This area was made a part of the expanded British West Florida colony.[31] The British changed the name of Fort Condé to Fort Charlotte, after Queen Charlotte.[32]

            The British were eager not to lose any useful inhabitants and promised religious tolerance to the French colonists; ultimately 112 French colonists remained in Mobile.[33] The first permanent Jewish settlers came to Mobile in 1763 as a result of the new British rule and religious tolerance. Jews had not been allowed to officially reside in colonial French Louisiana due to the Code Noir, a decree passed by France's King Louis XIV in 1685 that forbade the exercise of any religion other than Roman Catholicism, and ordered all Jews out of France's colonies. Most of these colonial-era Jews in Mobile were merchants and traders from Sephardic Jewish communities in Savannah, Georgia and Charleston, South Carolina; they added to the commercial development of Mobile.[34] In 1766 the total population was estimated to be 860, though the town's borders were smaller than during the French colonial period.[33] During the American Revolutionary War, West Florida and Mobile became a refuge for loyalists fleeing the other colonies.[35]

            While the British were dealing with their rebellious colonists along the Atlantic coast, the Spanish entered the war in 1779 as an ally of France. They took the opportunity to order Bernardo de Galvez, Governor of Louisiana, on an expedition east to retake West Florida.[36] He captured Mobile during the Battle of Fort Charlotte in 1780, as part of this campaign. The Spanish wished to eliminate any British threat to their Louisiana colony west of the Mississippi River, which they had received from France in the 1763 Treaty of Paris.[35] Their actions were condoned by the revolting American colonies, partially evidenced by the presence of Oliver Pollack, representative of the American Continental Congress. Due to strong trade ties, many residents of Mobile and West Florida remained loyal to the British Crown.[35][36] The Spanish renamed the fort as Fortaleza Carlota, and held Mobile as a part of Spanish West Florida until 1813, when it was seized by United States General James Wilkinson during the War of 1812.[37]</p>

    </div>

    </section>

  </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page section nav {

    display: block;

    position: sticky;

    overflow: hidden;

    background-color: rgb(189, 33, 186);

    padding: 20px;

    top: 0px;

  }

  .total-page section nav a {

    text-decoration: none;

    margin: 20px;

    color: rgb(252, 249, 249);

  }

  .total-page section nav i {

    color: rgb(249, 246, 246);

  }

  @media screen and (max-width: 789px) {

    .total-page section nav {

      flex-direction: column;

      align-items: flex-start;

      justify-content: flex-start;

      background-color: lightpink;

      color: #ffff;

    }

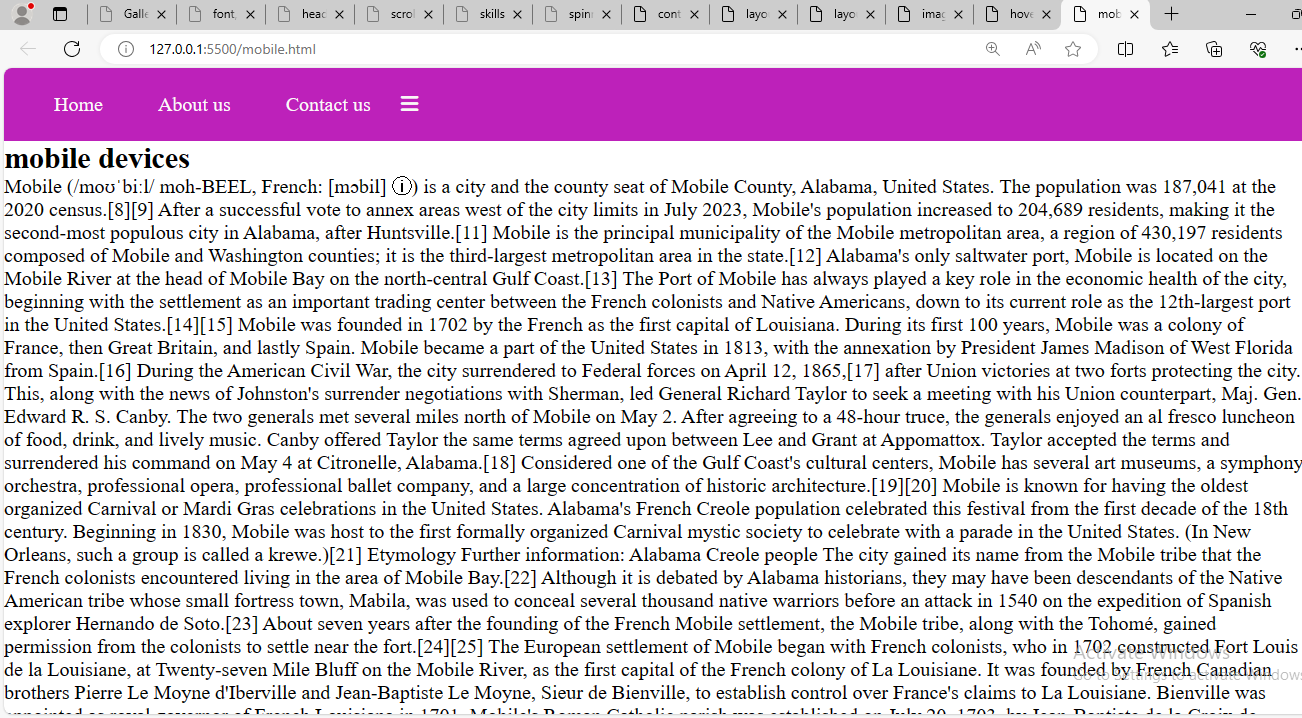
    .total-page section nav a {

      margin: 10px 20px;

    }

  }

**Output:**

****

1. Style a table to have alternating row colors and a hover effect on rows.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>hover rows</title>

    <link rel="stylesheet" href="rows.css">

</head>

<body>

    <div class="total-page">

        <table>

          <tr>

            <th>name</th>

            <th>number</th>

            <th>id</th>

          </tr>

          <tr>

            <td>sharvani</td>

            <td>9345676534</td>

            <td>A0579</td>

          </tr>

          <tr>

            <td>Raju</td>

            <td>7337070131</td>

            <td>A0578</td>

          </tr>

          <tr>

            <td>Swetha</td>

            <td>9087654321</td>

            <td>A0789</td>

          </tr>

        </table>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

  }

  table,

  th,

  td {

    border: 2px solid rgb(247, 7, 207);

    border-collapse: collapse;

    padding: 20px;

  }

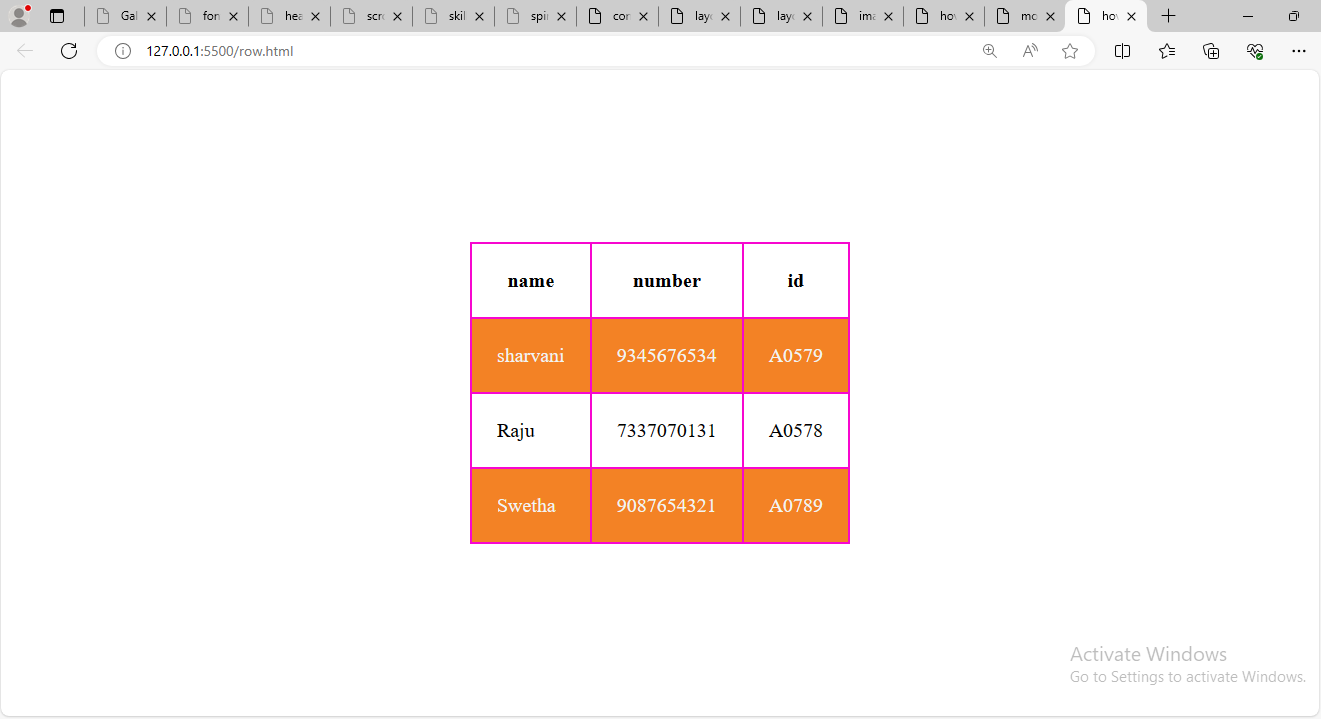
  table tr:nth-child(even) {

    background-color: rgb(243, 130, 37);

    color: rgb(241, 244, 243);

  }

**Output:**

****

1. Design a progress tracker with steps using HTML and CSS.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>steps</title>

    <link rel="stylesheet" href="steps.css">

</head>

<body>

    <div class="total-page">

        <div class="steps active">

          <span>database 1</span>

        </div>

        <div class="steps">

          <span>database 2</span>

        </div>

        <div class="steps">

          <span>database 3</span>

        </div>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page {

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

  }

  .total-page .steps span {

    padding: 20px 40px;

    margin: 10px;

    background-color: rgb(246, 247, 248);

    border: 2px solid rgb(224, 212, 213);

    box-shadow: 4px 4px 10px rgb(57, 56, 56);

    border-radius: 10px;

    color: rgb(7, 108, 122);

  }

  .total-page .steps .active span {

    background-color: lightpink;

    color: rgb(45, 44, 44);

  }

**Output:**

****

1. Create a parallax scrolling effect on a webpage background.

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>background</title>

    <link rel="stylesheet" href="back.css">

</head>

<body>

    <div class="total-page">

        <h2>mobile phones</h2>

        <div class="parallax"></div>

        <div class="content">

          <h1>mobile devices</h1>

          <p>we can use them anywhere</p>

        </div>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .parallax {

    background-image: url("images.jpeg");

    background-size: cover;

    background-attachment: fixed;

    height: 100%;

  }

  .content {

    height: 1000px;

    background-color: rgba(205, 137, 137, 0.8);

    padding: 20px;

  }

  .total-page{

    display: inline-block;

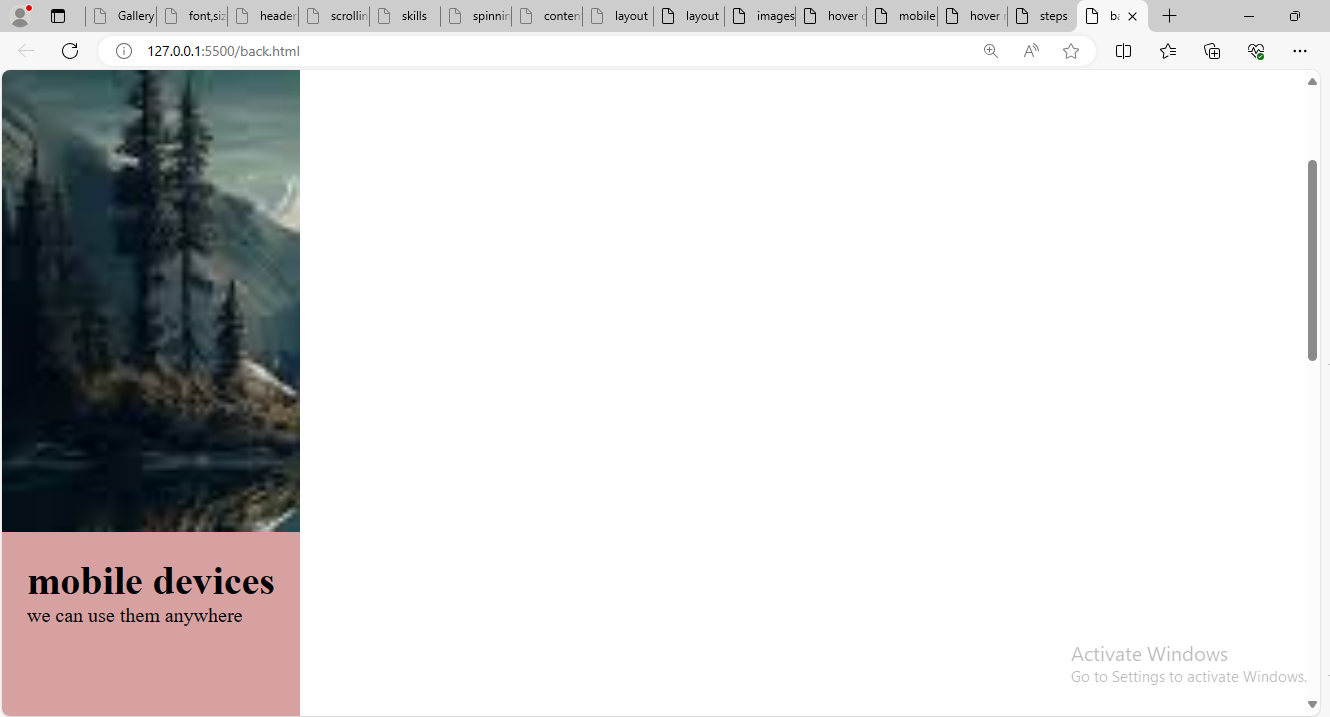
    justify-content: center;

    align-items: center;

    height: 100vh;

  }

**Output:**

****

1. Build a dropdown select menu with options styled using CSS

**Html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>food items</title>

    <link rel="stylesheet" href="app.css">

</head>

<body>

    <div class="total-page">

        <select>

          <option>select food from the list</option>

          <option>biriyani</option>

          <option>ice cream</option>

          <option>pani puri</option>

          <option>chat</option>

          <option>dhai papad</option>

          <option>curd rice</option>

          <option>mutton</option>

          <option>egg rice</option>

          <option>noddles</option>

          <option>pav bhaji</option>

          <option>white rice</option>

        </select>

      </div>

</body>

</html>

**Css:**

\* {

    margin: 0px;

    padding: 0px;

    box-sizing: border-box;

  }

  .total-page{

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

  }

  .total-page select {

    border: 3px solid lightseagreen;

    padding: 10px;

    width: 50%;

    border-radius: 20px;

    -webkit-appearance: none;

  }

  .total-page select option {

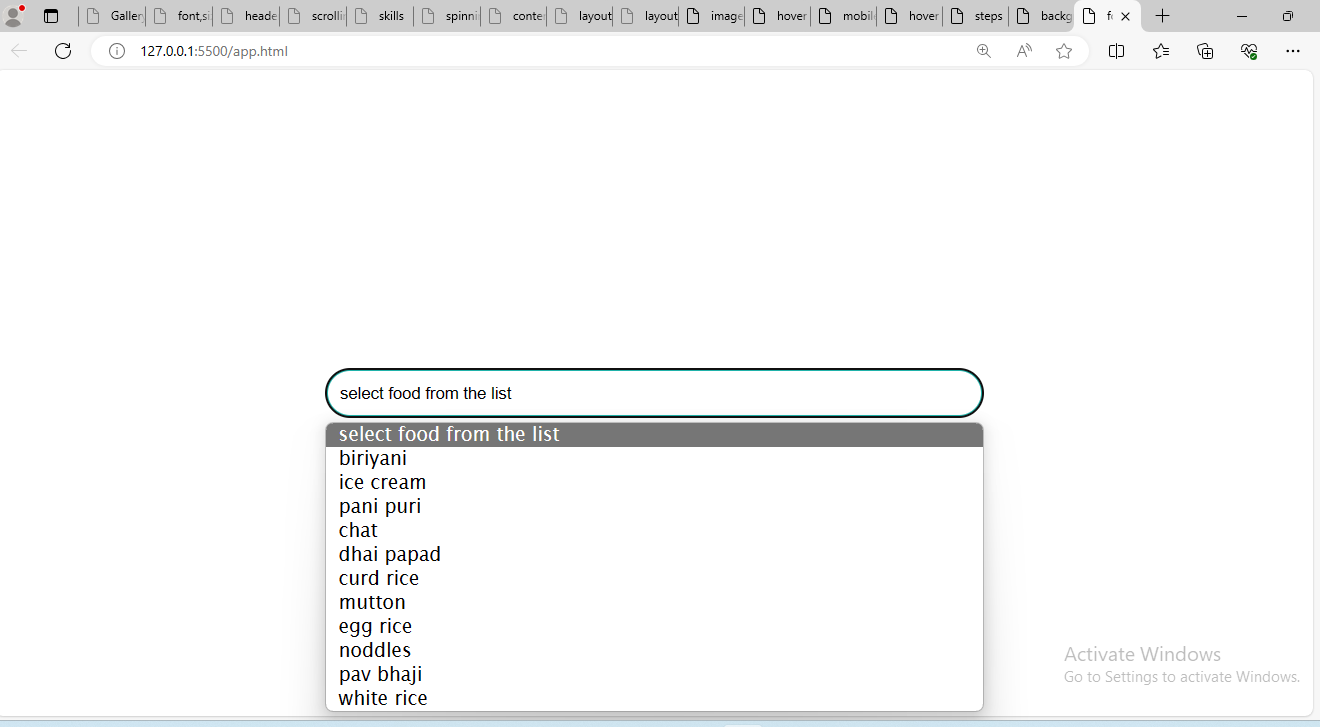
    font-size: 15px;

    font-family: "Lucida Sans", "Lucida Sans Regular", "Lucida Grande",

      "Lucida Sans Unicode", Geneva, Verdana, sans-serif;

  }

**Output:**

****

**mini projects:atleast need to complete 4 projects by monday**

1. Portfolio Website: Create a personal portfolio website showcasing your skills, projects, and contact information. Include sections for about me, projects, skills, and contact.
2. E-commerce Product Page: Design a product page for an online store. Include product images, description, price, and add-to-cart functionality.
3. Travel Blog: Build a travel blog website with articles, images, and categories for different destinations. Use CSS to style the layout and make it visually appealing.
4. Restaurant Website: Design a website for a restaurant with a menu, photo gallery, location/map, and contact form for reservations.
5. Event Landing Page: Create a landing page for an upcoming event or conference. Include event details, schedule, speakers, and registration form.
6. Online Resume/CV: Build an online resume or CV with sections for education, work experience, skills, and contact information. Use CSS to make it visually appealing and easy to read.
7. Weather App Interface: Design a simple weather app interface showing current weather conditions, temperature, and forecast for the week.
8. Fitness Tracker Dashboard: Create a dashboard for tracking fitness goals, workouts, and progress. Include charts/graphs to visualize data.
9. Music Player Interface: Design a music player interface with play/pause buttons, volume control, and playlist display.
10. Recipe Blog: Build a recipe blog website with recipes, ingredients, instructions, and cooking tips. Use CSS to style the layout and make it visually appealing.
11. Job Board: Create a job board website where employers can post job listings and job seekers can search and apply for jobs.