Color and Backgrounds, Flexbox, and Grid:

10. Color and Backgrounds:

Example Code:
```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Color and Backgrounds Example</title>
  <style>
    body {
      background-color: #f0f0f0; /* Set background color */
      color: #333; /* Set text color */
    }

    .container {
      background-color: #fff; /* Set background color for a container */
      padding: 20px;
      border-radius: 10px; /* Add rounded corners to the container */
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); /* Add a subtle box shadow */
    }
  </style>
</head>
<body>

  <div class="container">
    <h1>This is a container with a background color</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>
  </div>

</body>
</html>
```

This HTML document sets the background color for the entire page and a specific container. It also demonstrates setting the text color, adding rounded corners, and applying a subtle box shadow.

11. Flexbox:

Example Code:

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Flexbox Example</title>
  <style>
    .container {
      display: flex; /* Enable flexbox on the container */
      justify-content: space-between; /* Distribute items with space between them */
    }

    .item {
      width: 100px;
      height: 100px;
      background-color: #3498db;
      margin: 10px;
    }
  </style>
</head>
<body>

  <div class="container">
    <div class="item">1</div>
    <div class="item">2</div>
    <div class="item">3</div>
  </div>

</body>
</html>
```

This example uses flexbox to create a container with three items. The `justify-content: space-between;` property evenly distributes the items with space between them.

12. Grid:

Example Code:
```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grid Example</title>
  <style>
    .container {
      display: grid; /* Enable grid on the container */
      grid-template-columns: repeat(3, 1fr); /* Create three equal-width columns */
      gap: 10px; /* Add a 10px gap between grid items */
    }

    .item {
      background-color: #2ecc71;
      padding: 20px;
      text-align: center;
    }
  </style>
</head>
<body>

  <div class="container">
    <div class="item">1</div>
    <div class="item">2</div>
    <div class="item">3</div>
    <div class="item">4</div>
    <div class="item">5</div>
    <div class="item">6</div>
  </div>

</body>
</html>
```

This example uses CSS Grid to create a container with six items arranged in three columns. The `grid-template-columns` property sets equal-width columns, and `gap` adds spacing between the items.