(1)What are the benefits of using CSS?

➡ Ans:

These are the three major benefits of CSS:

(1) Solves a big problems

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: if you are developing a large website where fonts and color information are added on every single page, it will be become a long and expensive process. CSS was created to solve this problem.

(2) Saves a lot of time

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

(3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

(4) Flexibility

CSS provides a great deal of flexibility and control over the presentation of web content, allowing you to create complex and sophisticated layouts, and apply different styles to different elements of a page.

(2)What are the disadvantages of CSS?

➡ Ans:

(1)Dependency on external style sheets:

CSS is typically stored in external style sheets, which are accessed via links from a website's HTML. This means that if the style sheet is not loaded correctly or if the link to the style sheet is broken, the website may not display properly. This could be a problem if the style sheet takes a long time to load or your internet connection is slow.

(2)Compatibility issues with browsers:

Because various browsers interpret and render CSS differently, there might be inconsistencies and compatibility issues. Developers may need to spend more time and effort testing and debugging their CSS code in order to ensure that it runs correctly across different browsers.

(3)Confusion due to many CSS levels

Beginners are more vulnerable to this issue. They might get confused while opting to learn CSS as there are many levels of CSS such as CSS2, CSS3, etc.

(3) What is the difference between CSS2 and CSS3?

➡ Ans:

CSS2	CSS3
CSS2 was released in 1998.	While CSS3 is a more recent version that was first introduced in 1999 and has been updated periodically since then.
CSS does not support responsive design.	CSS3 is the latest version and supports the responsive design.
CSS is not divided into modules.	CSS3 could split into modules.
CSS cannot produce 3D animation and transformation.	In CSS3, the animation and 3D transformations are used.
In CSS, animations are written in JavaScript and JQuery. It has not to design layer features and page elements. It had no special effects such as shadowing text, text animation, etc.	In CSS3, the developer adds text-shadows to make it easy and effective. They add words for the visual effects of the break line and a comfortable fit inside the column. It changes the size and color of the text.
CSS provides unique color schemas and standard color.	CSS3 supports HSL RGBA, HSLA and the gradient colors.
CSS is slower.	CSS3 is faster than CSS.

(4) Name a few CSS style components?

➡ Ans:

- Color: You can specify the color of text, backgrounds, borders, and other elements using CSS.
- Font: You can set the font family, size, weight, and style of text using CSS.
- Box Model: The box model is a fundamental concept in CSS that defines how elements are laid out on the page. It includes properties like padding, margin, and border.
- Layout: You can control the layout of elements on the page using properties like display, position, and float.
- Responsive Design: You can create designs that adapt to different screen sizes using CSS media queries and responsive design techniques.
- Animation: You can create animations and transitions using CSS keyframes and animation properties.
- Flexbox: Flexbox is a layout model in CSS that allows you to create flexible and responsive layouts.
- Grid: CSS Grid is another layout model that allows you to create more complex and structured layouts than Flexbox.

- Backgrounds: You can set the background color or image of an element using CSS.

- Borders: You can add borders to elements using CSS, and control properties like border color, width, and style.

(5) What do you understand by CSS opacity?

➡ Ans:

CSS opacity is a property that controls the transparency of an element. It allows you to control the degree to which an element is transparent, where a value of 0 means the element is completely transparent (invisible), and a value of 1 means the element is completely opaque (fully visible).

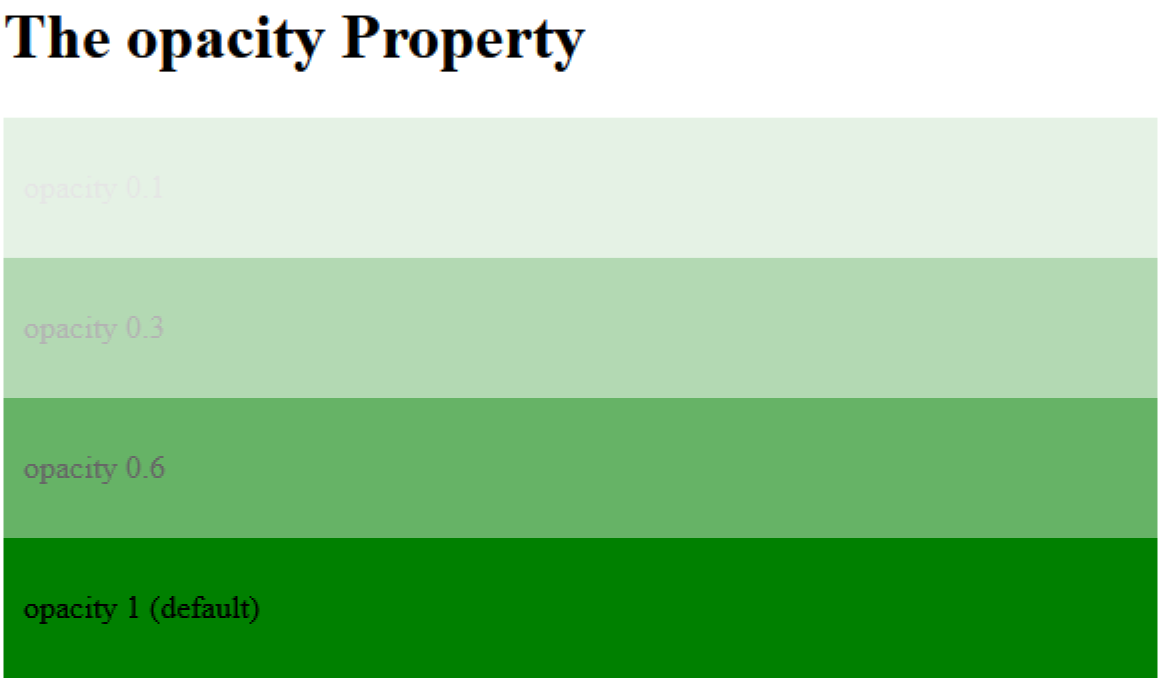
You can set the opacity of an element using the opacity property in CSS. The value for opacity ranges from 0 (completely transparent) to 1 (completely opaque).

For example, to make an element semi-transparent, you could set the opacity to a value between 0 and 1, like opacity: 0.5.

Example:

```
<!DOCTYPE html>

<html>

<head>

<style>

div {

  background-color: green;

  padding: 10px;
}
div.first {
  opacity: 0.1;
}

div.second {
  opacity: 0.3;
}
div.third {
  opacity: 0.6;
}
</style>
</head>
<body>

<h1>The opacity Property</h1>

<div class="first"><p>opacity 0.1</p></div>
<div class="second"><p>opacity 0.3</p></div>
<div class="third"><p>opacity 0.6</p></div>
<div><p>opacity 1 (default)</p></div>
</body>
</html>
```

Output:



The opacity Property

opacity 0.1

opacity 0.3

opacity 0.6

opacity 1 (default)

(6) How can the background color of an element be changed?

Ans:

The background color of an element can be changed using CSS

Cascading Style Sheets) by specifying the desired color value for

(Cascading Style Sheets) by specifying the desired color value for the background-color property of the element.

Here's an example:

```
<div style="background-color: red;">
        This is a red div </div>
```

In the above example, the background-color property is set to red, which changes the background color of the div element to red.

You can also specify the color using other formats like hexadecimal, RGB, RGBA, HSL, HSLA values, or use color names.

(7) How can image repetition of the backup be controlled?

➡ Ans:

To control the repetition of an image in the background, use the background-repeat property.

(8) What is the use of the background-position property?

➡ Ans:

The background-position property in CSS is used to set the position of a background image relative to its container.

It determines where the background image should be placed within the element's content box.

This property accepts two values, which represent the horizontal and vertical positions of the background image, respectively. The values can be specified in different units such as pixels, percentages, or keywords like "left," "center," "right," "top," "middle," and "bottom."

For example, background-position: center top; will center the background image horizontally and place it at the top of the container.

The background-position property is useful when you want to control the placement of a background image within an element.

(9) Which property controls the image scroll in the background?

➡ Ans:

The background-attachment property sets whether a background image scrolls with the rest of the page, or is fixed.

(10) Why should background and color be used as separate properties?

➡ Ans:

Background property:

The background property is typically used to set the background color or image of an element, such as a section, a div, or the entire body of a webpage. It can be used to create visual interest, add texture or patterns, or provide context for the content on the page.

Color property:

The color property is used to set the color of the text, borders, or other foreground elements within an element. It is primarily used to improve the readability and accessibility of the content on a page, making it easier for users to consume and interact with.

Overall, using background and color as separate properties provides designers with greater flexibility and control over the visual presentation of a webpage, making it easier to create engaging and accessible designs.

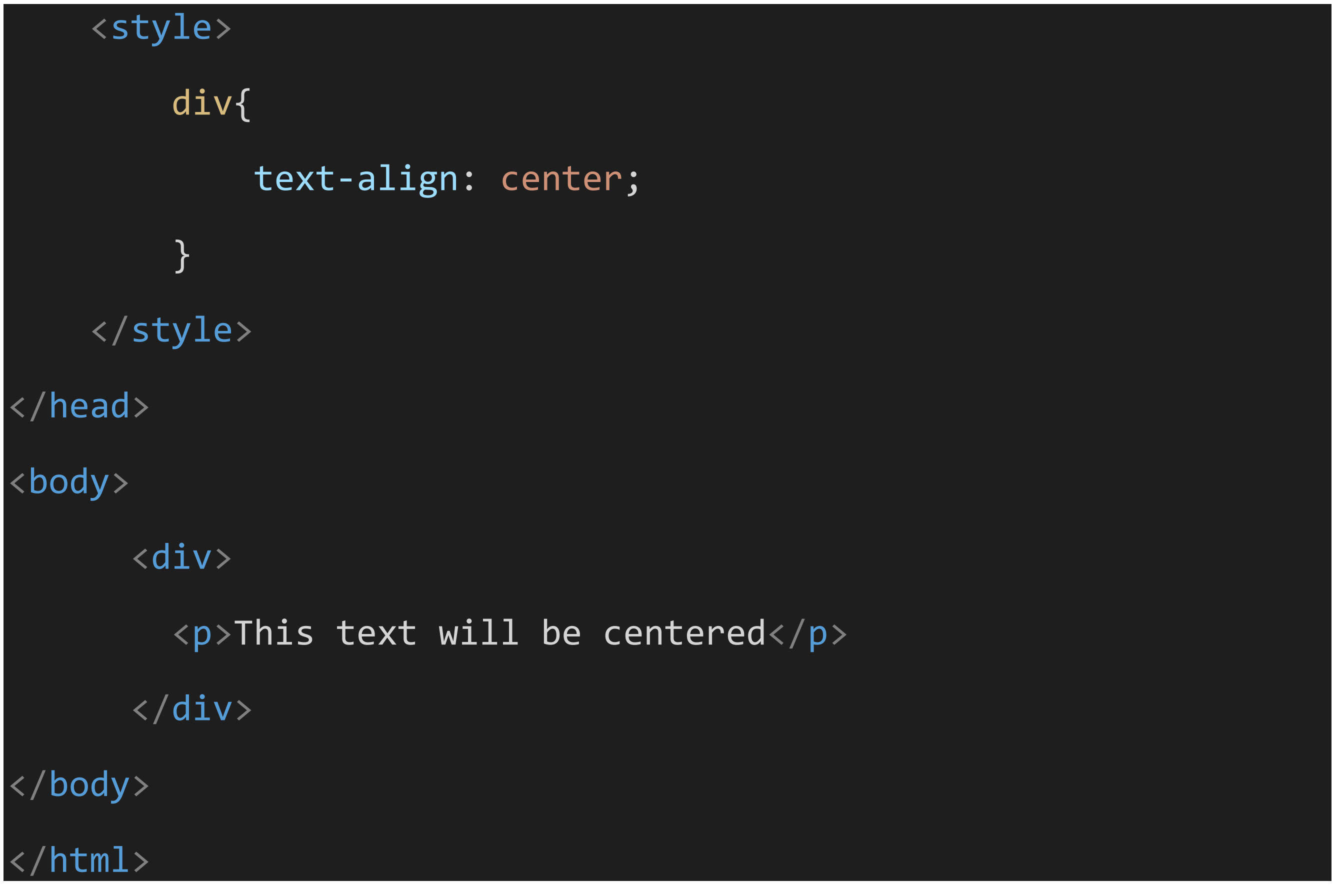
(11) How to center block elements using CSS1?

➡ Ans:

CSS1, also known as Cascading Style Sheets Level 1, was released in 1996 and did not provide any specific properties for centering block elements. However, there are a few ways to achieve centering using CSS1:

(1) Using text-align:

If the block element contains text or inline elements, you can center it horizontally by setting the text-align property of its parent element to center. Here's an example:



```
    <style>
        div{
            text-align: center;
        }
    </style>
</head>
<body>
      <div>
        <p>This text will be centered</p>
      </div>
</body>
</html>
```

In this example, the text-align property of the parent div element is set to center, which centers the text inside the p element.

(12)How to maintain the CSS specifications?

- Ans:

[1]Stay up-to-date: Keep track of updates to the CSS specification and stay current with the latest version. The World Wide Web Consortium (W3C) is the primary organization responsible for developing and maintaining the CSS standard, and you can find the latest version of the CSS specification on their website.

[2]Follow best practices: Use best practices for writing CSS code, such as following naming conventions, organizing your code for readability, and commenting your code to explain what each section does.

[3]Test your code: Test your CSS code thoroughly to ensure that it works correctly in all browsers and devices. This includes testing for compatibility with older browsers and ensuring that your code is accessible to users with disabilities.

[4]Validate your code: Use a CSS validator to check your code for errors and ensure that it adheres to the CSS specification. The W3C provides a CSS validator that you can use to check your code for compliance.

[5]Continuously improve: Continuously improve your CSS skills and knowledge by reading blogs and articles, participating in online communities, attending conferences and workshops, and staying up-to-date with new developments and trends in the field.

(13) What are the ways to integrate CSS as a web page?

➡ Ans:

There are several ways to integrate CSS (Cascading Style Sheets) into a web page. Here are some of the most common ways:

- External CSS:

In this method, the CSS code is written in a separate file with a .css extension, and it is linked to the HTML document using the <link> tag.
This is the most recommended method for CSS integration, as it separates the presentation layer from the content and structure layer of the web page.

```
Ex: <head>
          <link  rel=”stylesheet”  href=”style.css” >
    </head>
```

- Internal CSS:

In this method, the CSS code is written within the <style> tags in the head section of the HTML document.
This method is useful for small websites or pages with a simple layout.

```
Ex: <head>
          <style>
          body { background-color : black; }
          p { color : red; }
          </style>
    </head>
```

- Inline CSS:

In this method, the CSS code is written directly in the HTML tag using the style attribute.

This method is useful for making quick style changes to a particular element on a web page.

Ex: <p style=”color : blue;”>This is a paragraph.</p>

(14) What is an embedded style sheet?

➡ Ans:

- Embedded style sheets are a way of adding style information to a web page by including CSS (Cascading Style Sheets) code directly in the head section of an HTML document. This is done using the <style> tag, which allows developers to define styles for specific elements or classes within the same HTML file.
- Embedded style sheets are often used for small or simple websites because they keep the style information in the same file as the HTML content, making it easier to manage and update. They are also useful for making quick changes to the style of a page without having to edit an external CSS file.
- However, embedded style sheets can become unwieldy and difficult to manage for larger websites with many pages, as it can be difficult to keep track of all the styles in a single file. In these cases, it is often better to use an external CSS file that can be linked to from multiple HTML pages.

(15) What are the external style sheets?

➡ Ans:

External style sheets are separate files containing CSS (Cascading Style Sheets) code that define the visual appearance of a web page. They are used to separate the presentation of a web page from its content, making it easier to maintain and update the styling across multiple pages.

To use an external style sheet, the web page needs to link to it using the <link> element in the <head> section of the HTML document. The <link> element specifies the location of the external style sheet file and establishes the relationship between the HTML document and the CSS file.

Using external style sheets also allows for the separation of concerns, where designers can focus on the visual aspects of the website without interfering with the HTML structure and content, which can be handled by developers or content creators.

(16) What are the advantages and disadvantages of using external style sheets?

➡ Ans:

<u>Advantages:</u>

1. Consistency: External style sheets allow you to keep the styling of your website consistent across all pages. You can define a set of rules for fonts, colors, and other design elements in a single file and apply them to multiple pages.
2. Easy Maintenance: When you need to make a change to the styling of your website, you only have to make the change once in the external style sheet. This makes it easier to maintain and update your website over time.

3. Additionally, it keeps your other pages code nice and tidy as all the styling, which can often entail a lot of code, is separated onto a completely different file.

Disadvantages:

1. Your pages may not be rendered correctly until the external CSS is loaded. Uploading or linking to multiple CSS files can increase your site's download time.
2. Increased HTTP requests: External style sheets require an additional HTTP request to download the file, which can increase the page load time.
3. some older browsers may not support certain CSS features, which can cause the styling to break.

(17) What is the meaning of the CSS selector?

➡ Ans:

CSS selectors use a variety of syntax to select elements based on their tag name, class, ID, attributes, or their relationship to other elements on the page. Here are a few examples:

Element selector : selects all elements of a certain type, such as all <p> elements.

Class selector : selects all elements with a particular class attribute, such as <div class="my-class">.

ID selector : selects a single element with a unique ID attribute, such as <div id="my-id">.

Attribute selector : selects elements based on the presence or value of a specific attribute, such as <input type="checkbox">.

Descendant selector: selects elements that are descendants of a particular element, such as <ul> li.

(18) What are the media types allowed by CSS?

- ➡ Ans:

In CSS, there are several media types that can be used to apply styles to different devices and environments.

The media types are:

For mobile : 320px to 480px

For Tablets, i-pad : 481px to 768px

For small screen, laptops : 769px to 1024px

For Desktop, large screen : 1025px to 1200px

For Extra large screen, TV : 1201px and more

(19) What is the rule set?

- ➡ Ans:

In CSS (Cascading Style Sheets), the rule set refers to the syntax used to define the styles for a specific HTML element or a group of elements.

The basic syntax of a CSS rule set consists of a selector and a set of properties and values enclosed in curly braces. The selector specifies the HTML element(s) to which the styles should be applied, while the properties and values define the specific styles to be applied to those elements.

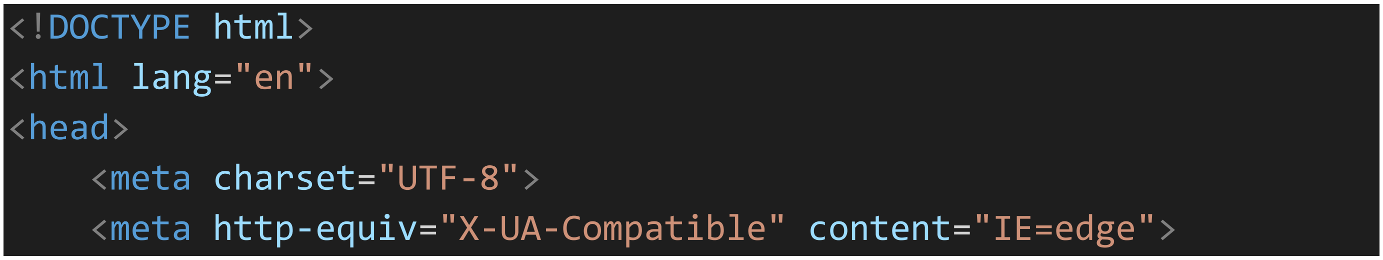
Here's an example of a CSS rule set:

```
h1{

    color: blue;

    font-size: 28px;

    text-align :center;

}
```

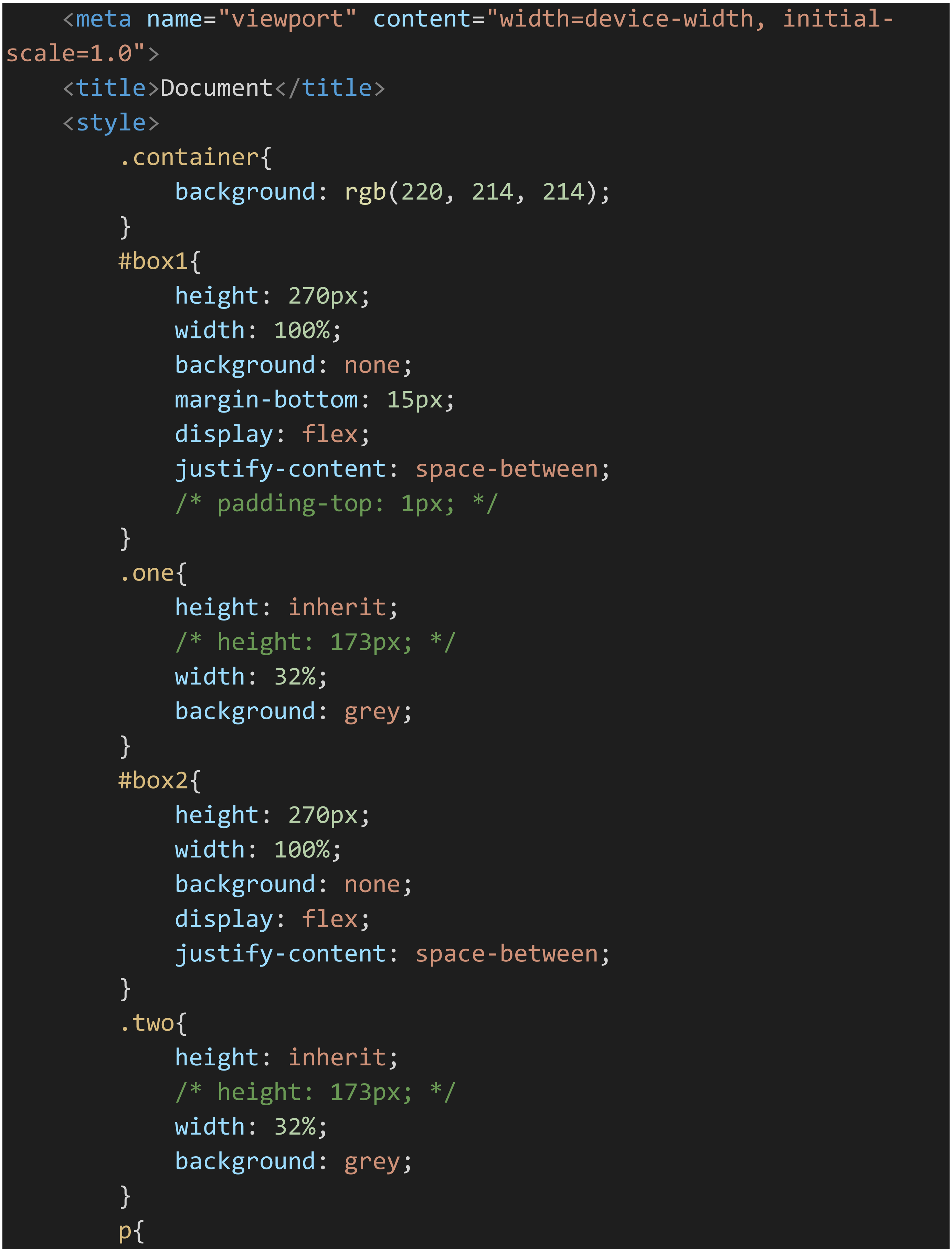
In this example, the selector is "h1", which targets all the heading level 1 elements in the HTML document. The properties and values inside the curly braces define the styles to be applied to those elements. In this case, the color of the text will be blue, the font size will be 28 pixels, and the text will be centered.

(20) Create Layouts

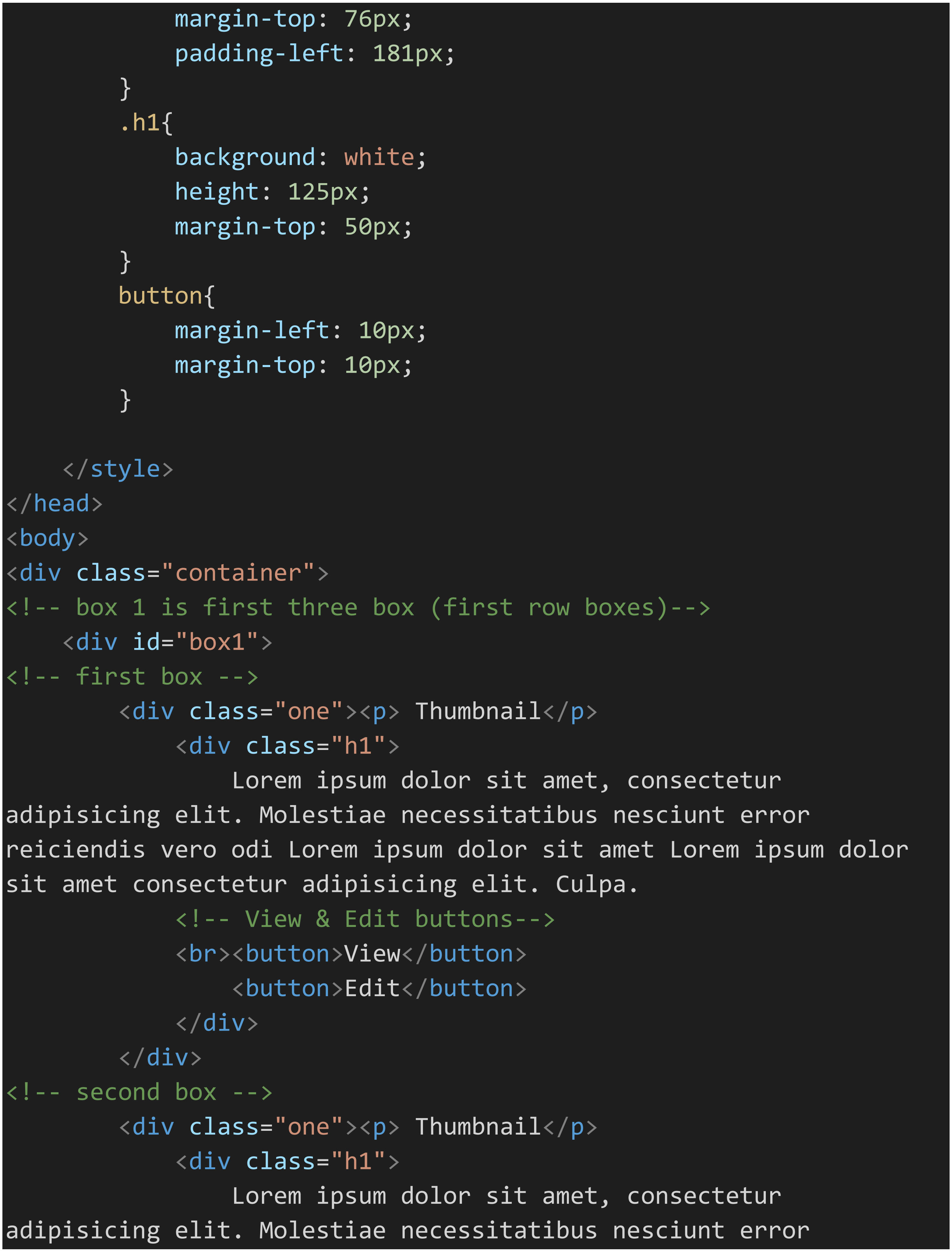
- Ans: Code,



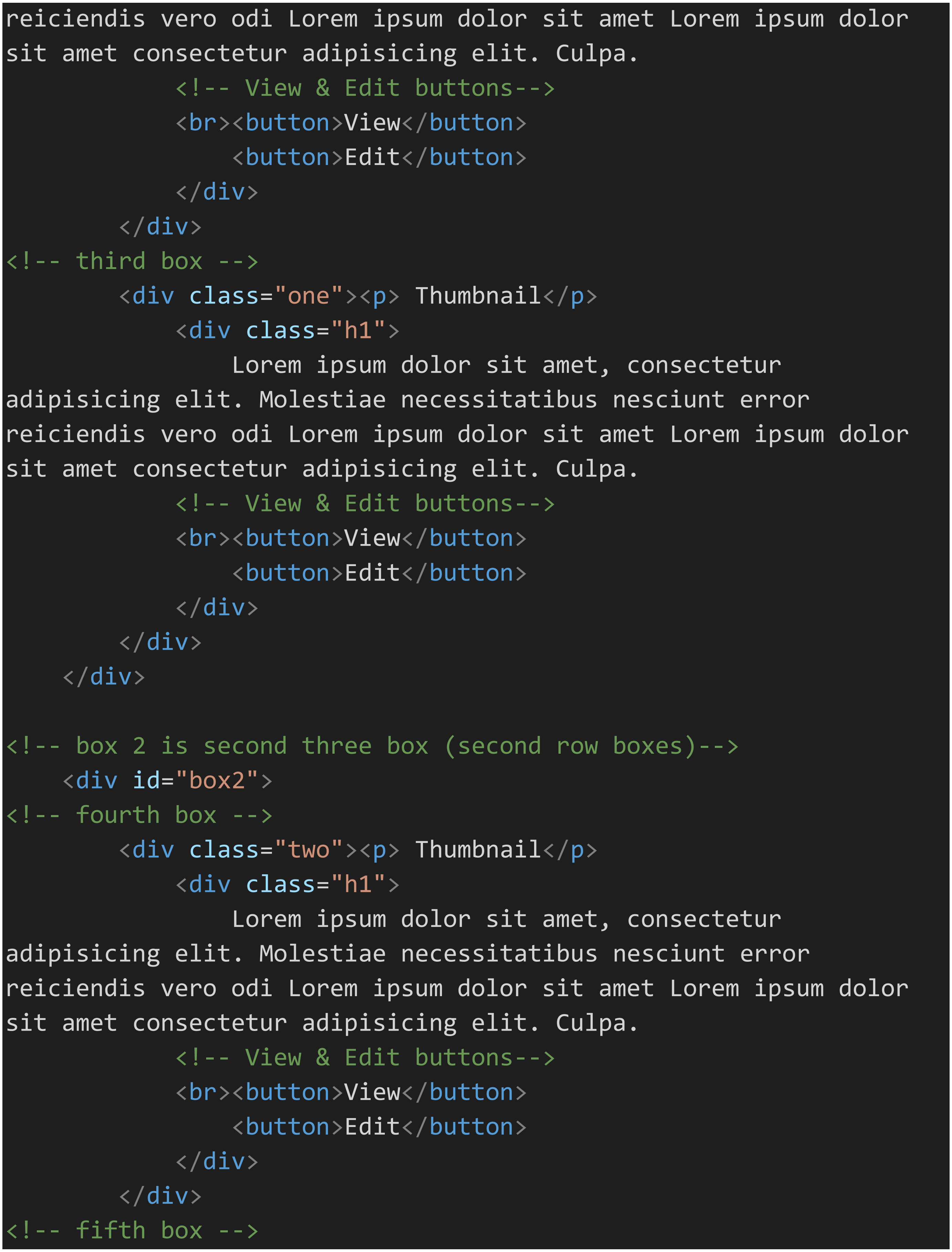
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```



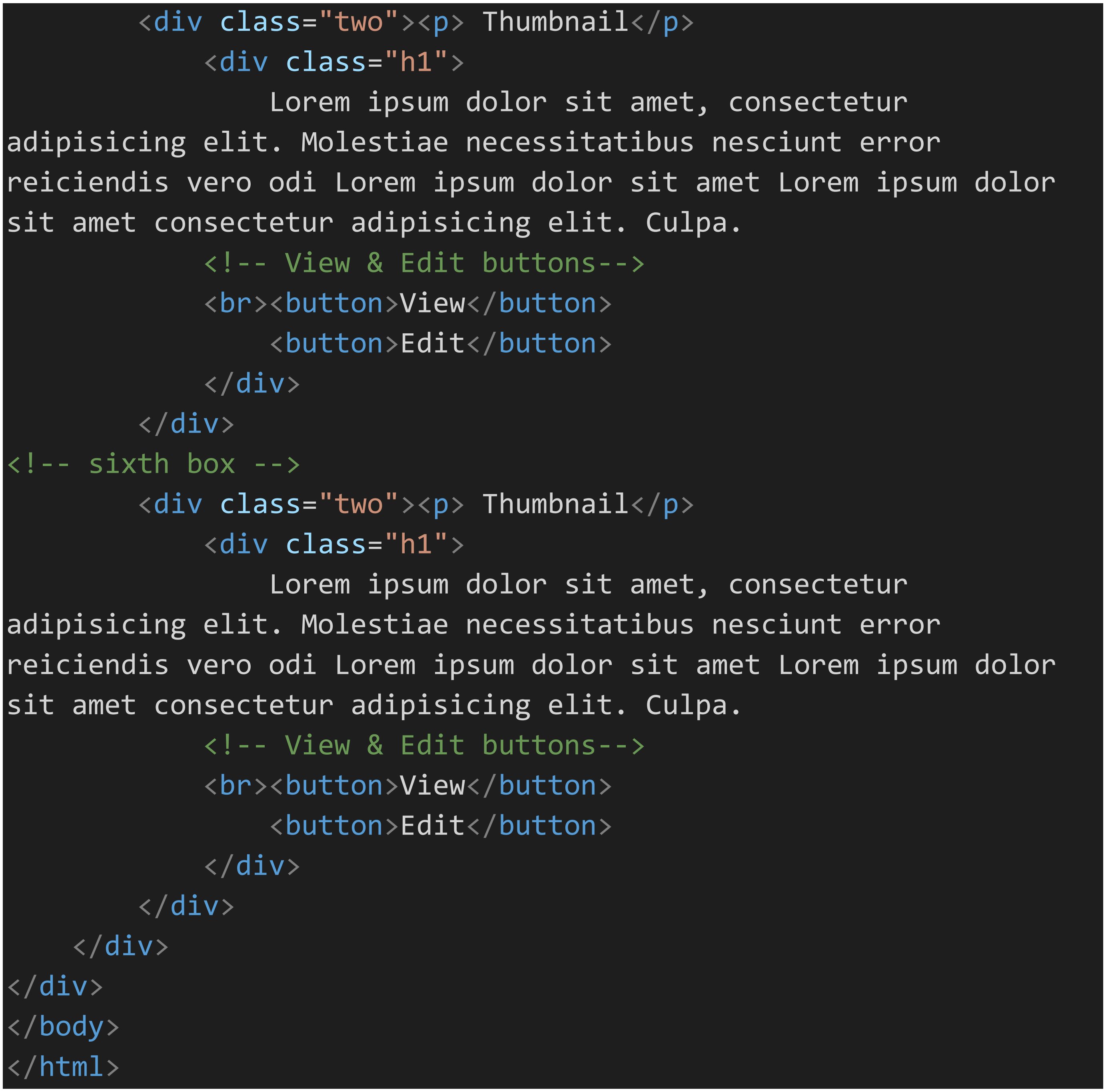
```
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Document</title>
    <style>
        .container{
            background: rgb(220, 214, 214);
        }
        #box1{
            height: 270px;
            width: 100%;
            background: none;
            margin-bottom: 15px;
            display: flex;
            justify-content: space-between;
            /* padding-top: 1px; */
        }
        .one{
            height: inherit;
            /* height: 173px; */
            width: 32%;
            background: grey;
        }
        #box2{
            height: 270px;
            width: 100%;
            background: none;
            display: flex;
            justify-content: space-between;
        }
        .two{
            height: inherit;
            /* height: 173px; */
            width: 32%;
            background: grey;
        }
        p{
```



```
            margin-top: 76px;
            padding-left: 181px;
        }
        .h1{
            background: white;
            height: 125px;
            margin-top: 50px;
        }
        button{
            margin-left: 10px;
            margin-top: 10px;
        }

    </style>
</head>
<body>
<div class="container">
<!-- box 1 is first three box (first row boxes)-->
    <div id="box1">
<!-- first box -->
        <div class="one"><p> Thumbnail</p>
            <div class="h1">
                Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Molestiae necessitatibus nesciunt error
reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor
sit amet consectetur adipisicing elit. Culpa.
            <!-- View & Edit buttons-->
            <br><button>View</button>
                <button>Edit</button>
            </div>
        </div>
<!-- second box -->
        <div class="one"><p> Thumbnail</p>
            <div class="h1">
                Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Molestiae necessitatibus nesciunt error
```

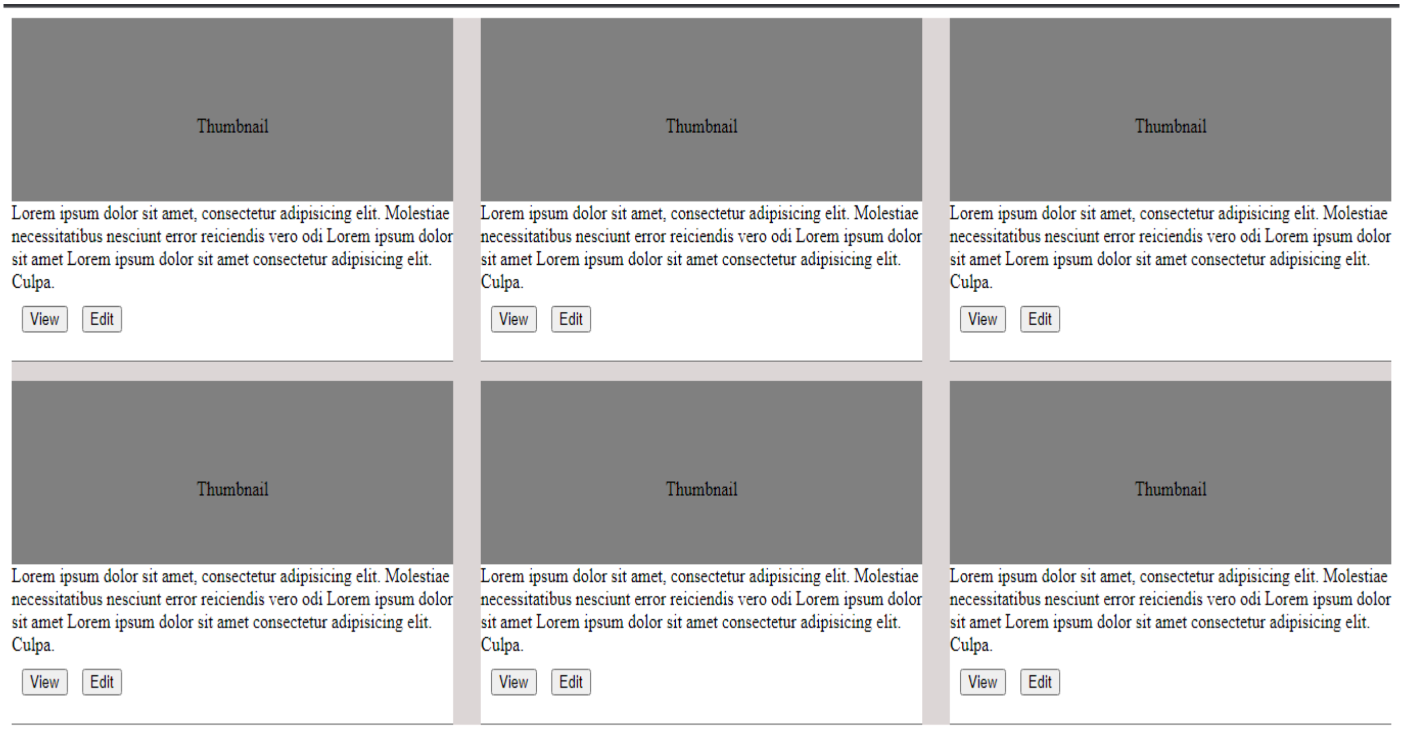


```
reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor
sit amet consectetur adipisicing elit. Culpa.
            <!-- View & Edit buttons-->
            <br><button>View</button>
                <button>Edit</button>
            </div>
        </div>
<!-- third box -->
        <div class="one"><p> Thumbnail</p>
            <div class="h1">
                Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Molestiae necessitatibus nesciunt error
reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor
sit amet consectetur adipisicing elit. Culpa.
            <!-- View & Edit buttons-->
            <br><button>View</button>
                <button>Edit</button>
            </div>
        </div>
    </div>

<!-- box 2 is second three box (second row boxes)-->
    <div id="box2">
<!-- fourth box -->
        <div class="two"><p> Thumbnail</p>
            <div class="h1">
                Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Molestiae necessitatibus nesciunt error
reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor
sit amet consectetur adipisicing elit. Culpa.
            <!-- View & Edit buttons-->
            <br><button>View</button>
                <button>Edit</button>
            </div>
        </div>
<!-- fifth box -->
```



```
        <div class="two"><p> Thumbnail</p>
            <div class="h1">
                Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Molestiae necessitatibus nesciunt error
reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor
sit amet consectetur adipisicing elit. Culpa.
            <!-- View & Edit buttons-->
            <br><button>View</button>
                <button>Edit</button>
            </div>
        </div>
<!-- sixth box -->
        <div class="two"><p> Thumbnail</p>
            <div class="h1">
                Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Molestiae necessitatibus nesciunt error
reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor
sit amet consectetur adipisicing elit. Culpa.
            <!-- View & Edit buttons-->
            <br><button>View</button>
                <button>Edit</button>
            </div>
        </div>
    </div>
</div>
</body>
</html>
```

Output:



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Molestiae necessitatibus nesciunt error reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor sit amet consectetur adipisicing elit. Culpa.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Molestiae necessitatibus nesciunt error reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor sit amet consectetur adipisicing elit. Culpa.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Molestiae necessitatibus nesciunt error reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor sit amet consectetur adipisicing elit. Culpa.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Molestiae necessitatibus nesciunt error reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor sit amet consectetur adipisicing elit. Culpa.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Molestiae necessitatibus nesciunt error reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor sit amet consectetur adipisicing elit. Culpa.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Molestiae necessitatibus nesciunt error reiciendis vero odi Lorem ipsum dolor sit amet Lorem ipsum dolor sit amet consectetur adipisicing elit. Culpa.