# Assignment2

# Module (CSS and CSS 3) 2

# 1)What are the benefits of using CSS?

**Ans**.

* The main advantage of CSS is that style is applied consistently across variety of sites. One instruction can control several areas which is advantageous.
* Web designers needs to use few lines of programming for every page improving site speed.
* Cascading sheet not only simplifies website development, but also simplifies the maintenance as a change of one line of code affects the whole web site and maintenance time.
* It is less complex therefore the effort are significantly reduced.
* It helps to form spontaneous and consistent changes.
* CSS changes are device friendly. With people employing a batch of various range of smart devices to access websites over the web, there’s a requirement for responsive web design.
* It has the power for re-positioning. It helps us to determine the changes within the position of web elements who are there on the page.
* These bandwidth savings are substantial figures of insignificant tags that are indistinct from a mess of pages.
* Easy for the user to customize the online page
* It reduces the file transfer size.

**2)What are the disadvantages of CSS?**

**Ans.**

* With CSS, what works with one browser might not always work with another. The web developers need to test for compatibility, running the program across multiple browsers.
* There exists a scarcity of security.
* After making the changes we need to confirm the compatibility if they appear. The similar change affects on all the browsers.
* The programing language world is complicated for non-developers and beginners. Different levels of CSS i.e. CSS, CSS 2, CSS 3 are often quite confusing.
* Browser compatibility (some styles sheet are supported and some are not).
* CSS works differently on different browsers. IE and Opera supports CSS as different logic.
* There might be cross-browser issues while using CSS.
* There are multiple levels which creates confusion for non-developers and beginners.

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

**Ans.**

So we’ve established that CSS3 is just the latest version of CSS. That’s terrific, but what sets it apart from its predecessor? The following table clearly illustrates the differences between CSS vs. CSS3.

|  |  |  |
| --- | --- | --- |
|  | **CSS** | **CSS3** |
| **Media Queries** | Doesn’t support | Supports responsive web design |
| **Browser Support** | No support for modern browsers, but it still works on older versions of Explorer or Chrome | Supported fully by all modern browsers |
| **compatibility BetweCen Versions** | Not compatible with CSS3 | Backward compatible with CSS |
| **Block Support** | Supports single blocks only | Supports multi-column text blocks |
| **Animation Use** | It only allows basic animations and doesn't support transformation, text animation, transition, or 3D animations | It offers advanced animations and many customization options. It also supports text animation, transformation, and transition |
| **Responsive Design** | It doesn’t support media queries, thus not ideal for making responsive designs | Works with media queries, thus allowing responsive web design |
| **Module Use** | It doesn’t have modules | Can group CSS codes into convenient modules |
| **Color Format** | It uses an old standard color format | It offers different gradient colors and schemes like RGBA, HSLA, HSL, etc. |
| **Performance** | It provides average performance and requires high memory usage | It offers fast, excellent performance and doesn’t use as much memory |

**4)Name a few CSS style components**

**Ans.**

The components of css style are:

**1.Selecter**:HTML element name, id name, class name.

**2.Property:**It's like an attribute such as background color,font-size,position,text-align,color,border etc.

**3.Values**:which defines property or values allocate for properties.

**5) What do you understand by CSS opacity?**

**Ans.**

The CSS opacity property is used to specify the transparency of an element. In simple word, you can say that it specifies the clarity of the image.

In technical terms, Opacity is defined as degree in which light is allowed to travel through an object.

**Example**

<!DOCTYPE html>

<html>

<head>

<style>

div {

background-color: #4CAF50;

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>

<p>The opacity property adds transparency to the background of an element, and to all of its child elements as well. This makes the text inside a transparent element hard to read:</p>

<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>

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

**Ans**.

To add background color in HTML, use the CSS background-color property. Set it to the color name or code you want and place it inside a style attribute. Then add this style attribute to an HTML element, like a table, heading, div, or span tag.

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-color: rgb(201, 76, 76);

}

</style>

</head>

<body>

<p>The background color can be specified with an RGB value.</p>

</body>

</html>

**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. You can use no-repeat value for the background-repeat property if you do not want to repeat an image, in this case, the image will display only once.

## Example

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("paper.gif");

background-repeat: repeat;

}

</style>

<body>

<h1>The background-repeat Property</h1>

<p>Here, the background image is repeated both vertically and horizontally. This is default.</p>

</body>

</html>

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

**Ans.**

The background-position property sets the starting position of a background image.

Tip: By default, a [background-image](https://www.w3schools.com/cssref/pr_background-image.php) is placed at the top-left corner of an element, and repeated both vertically and horizontally.

* syntex of the different values of the background-position property.
* **Keyword values**
* background-position: top;
* background-position: bottom;
* background-position: left;
* background-position: right;
* background-position: center;
* <**percentage> values**
* background-position: 25% 75%;
* **<length> values**
* background-position: 0 0;
* background-position: 1cm 2cm;
* background-position: 10ch 8em;
* **Multiple images**
* background-position: 0 0, center;
* **Edge offsets values**
* background-position: bottom 10px right 20px;
* background-position: right 3em bottom 10px;
* background-position: bottom 10px right;
* background-position: top right 10px;
* **Global values**
* background-position: inherit;
* background-position: initial;
* background-position: revert;
* background-position: revert-layer;
* background-position: unset;

**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.

**stntax**

background-attachment: scroll|fixed|local|initial|inherit;

**example**

<html>

<head>

<style>

.fixed-bg {

background-image: url("img\_tree.gif");

min-height: 500px;

background-attachment: fixed;

background-position: center;

background-repeat: no-repeat;

background-size: cover;

}

</style>

</head>

<body>

<p>In this example, we have created a fixed background image that will disappear slowly on scroll. Scroll the page to see the effect. <strong>Note:</strong> Try to remove the background-attachment property to really understand this example.</p>

<div class="fixed-bg"></div>

<div style="height:800px;background-color:yellow;">This div is only here to enable scrolling (height = 800 pixels).</div>

</body>

</html>

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

**Ans.**

if they should always be set together? There are two reasons behind this: It enhances the legibility of style sheets. The background property is a complex property in CSS, and if it is combined with color, the complexity will further increase.

**11) How to center block elements using CSS1?**

**Ans.**

We have seen that how this block element behaves, we observe that as they take full line width, to center them we only have the [margin](https://www.geeksforgeeks.org/css-margins-padding/) property that is controlling them horizontally.

The margin can control the position of the block element both horizontally and vertically. To center them, we can adjust the margin property such that it is placed in the center.

**Example**

<!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" />

    <style>

        h2,

        p {

            text-align: center;

        }

        .myblock {

            margin: auto;

            border: 2px solid red;

            width: fit-content;

            padding: 15px;

            text-align: center;

            background-color: lightyellow;

        }

        header {

            font-size: 40px;

            background-color: lightgreen;

            margin: auto;

            width: fit-content;

        }

        .myinline {

            padding: 10px;

            border: 2px solid blue;

        }

        .holder {

            text-align: center;

        }

    </style>

</head>

<body>

    <h2>Welcome To GFG</h2>

    <p>Default code has been loaded into the Editor.</p>

    <header>hello</header>

    <div class="myblock">

        div who has default display : block

    </div>

    <div class="holder">

        <div style="display: inline-block" class="myinline">

            inline block paragraph 1

        </div>

        <div style="display: inline-block" class="myinline">

            inline block paragraph 2

        </div>

    </div>

</body>

</html>

**12) How to maintain the CSS specifications?**

**Ans.**

The CSS specifications are maintained by the World Wide Web Consortium (W3C).

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

**Ans.**

There are three ways to integrate CSS into a Web page

**1.Inline:** HTML elements may have CSS applied to them via the STYLE attribute.

**2. Embedded:** By placing the code in a STYLE element within the HEAD element.

**3. Linked/ Imported**: Place the CSS in an external file and link it via a link element.

**13) What is embedded style sheets?**

**Ans.**

 It allows you to define styles for a particular HTML document as a whole in one place. This is done by embedding the **<style></style>** tags containing the CSS properties in the head of your document. Embedded style sheets are particularly useful for HTML documents that have unique style requirements from the rest of the documents in your project. However, if the styles need to be applied across multiple documents, you should link to an external style sheet instead of using individual embedded style sheets. Using embedded stylesheets holds a distinct advantage over inline styles which only allow you to address one HTML element at a time.

**Syntax:**The CSS syntax for embedded style sheets is exactly the same as other CSS code, apart from the fact that it is now wrapped within the <style></style> tags. The <style> tag takes the ‘type’ attribute that defines the type of style sheet being used (ie. text/CSS).

**Example 1:** Below is an HTML document with the CSS styling for the entire web page enclosed within the <style></style> tags. These properties would be applied to all corresponding elements in the HTML document.

<!DOCTYPE html>

<html>

<head>

    <title>Page Title</title>

    <!-- Embedded stylesheet -->

    <style>

        h2 {

            font-size: 1.5rem;

            color: #2f8d46;

            text-align: center;

        }

        p {

            font-variant: italic;

        }

    </style>

</head>

<body>

    <h2>Welcome To GFG</h2>

    <p>This document is using an embedded stylesheet!</p>

    <p>This is a paragraph</p>

    <p>This is another paragraph</p>

</body>

</html>

**14) What are the external style sheets?**

**Ans.**

To apply a rule to multiple pages, an external style sheet is used. An external style sheet is a separate CSS file that can be accessed by creating a link within the head section of the webpage. Multiple webpages can use the same link to access the stylesheet.

The link to an **external** style sheet is placed within the head section of the page.

<head>

<link rel="stylesheet" type="text/css" href="mystyle.css">

</head>

The actual style sheet file will contain CSS rules that are then applied across the entire page. For example:

body

{ background-color: ghostwhite;}

h1 { color: blue; font-size: 20px; font-family: verdana; font-style:italic;}

**15) What are the advantages and disadvantages of using external style sheets?**

**Ans.**

The advantages of External Style Sheets are as follows :

* With the help of External Style Sheets, the styles of numerous documents can be organized from one single file.
* In External Style Sheets, Classes can be made for use on numerous HTML element types in many forms of the site.
* In complex contexts, Methods like selector and grouping can be implemented to apply styles.

The disadvantages of External Style Sheets are as follows :

* An extra download is essential to import style information for each file.
* The execution of the file may be deferred till the external style sheet is loaded.
* While implementing style sheets, we need to test Web pages with multiple browsers in order to check compatibility issues.

**16) What is the meaning of the CSS selector?**

**Ans.** A CSS selector selects the HTML element(s) you want to style

We can divide CSS selectors into five categories:

* Simple selectors (select elements based on name, id, class)
* [Combinator selectors](https://www.w3schools.com/css/css_combinators.asp) (select elements based on a specific relationship between them)
* [Pseudo-class selectors](https://www.w3schools.com/css/css_pseudo_classes.asp) (select elements based on a certain state)
* [Pseudo-elements selectors](https://www.w3schools.com/css/css_pseudo_elements.asp) (select and style a part of an element)
* [Attribute selectors](https://www.w3schools.com/css/css_attribute_selectors.asp) (select elements based on an attribute or attribute value)

## 1)The CSS element Selector

* The element selector selects HTML elements based on the element name.

**example**

<!DOCTYPE html>

<html>

<head>

<style>

p {

text-align: center;

color: red;

}

</style>

</head>

<body>

<p>Every paragraph will be affected by the style.</p>

<p id="para1">Me too!</p>

<p>And me!</p>

</body>

</html>

## 2)The CSS id Selector

* The id selector uses the id attribute of an HTML element to select a specific element.
* The id of an element is unique within a page, so the id selector is used to select one unique element!
* To select an element with a specific id, write a hash (#) character, followed by the id of the element.

**example**

<!DOCTYPE html>

<html>

<head>

<style>

#para1 {

text-align: center;

color: red;

}

</style>

</head>

<body>

<p id="para1">Hello World!</p>

<p>This paragraph is not affected by the style.</p>

</body>

</html>

## 2)The CSS class Selector

* The class selector selects HTML elements with a specific class attribute.
* To select elements with a specific class, write a period (.) character, followed by the class name.
* example

<!DOCTYPE html>

<html>

<head>

<style>

.center {

text-align: center;

color: red;

}

</style>

</head>

<body>

<h1 class="center">Red and center-aligned heading</h1>

<p class="center">Red and center-aligned paragraph.</p>

</body>

</html>

## 3)The CSS Universal Selector

The CSS rule below will affect every HTML element on the page:

**example**

<!DOCTYPE html>

<html>

<head>

<style>

\* {

text-align: center;

color: blue;

}

</style>

</head>

<body>

<h1>Hello world!</h1>

<p>Every element on the page will be affected by the style.</p>

<p id="para1">Me too!</p>

<p>And me!</p>

</body>

</html>

## 4)The CSS Grouping Selector

* The grouping selector selects all the HTML elements with the same style definitions.
* Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

**example**

<!DOCTYPE html>

<html>

<head>

<style>

h1, h2, p {

text-align: center;

color: red;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<h2>Smaller heading!</h2>

<p>This is a paragraph.</p>

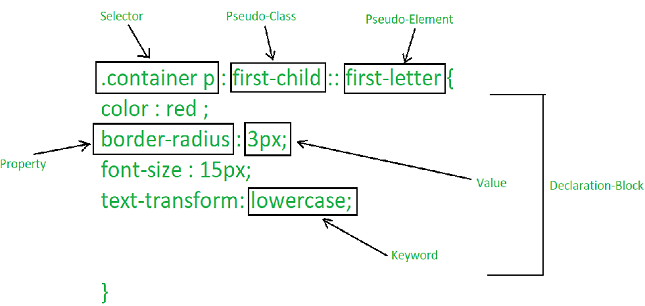
</body>

</html>

**15) What is the rule set?**

**Ans.** A CSS ruleset is various affirmations to various pieces or elements of the document. The objective is to apply a bunch of properties for certain distinct qualities to a solitary, or a particular arrangement of components in the connected HTML page.

**Visualization of CSS Ruleset**

. 

**16) Create Layouts**

## Ans,

## 1)Header

A header is usually located at the top of the website (or right below a top navigation menu). It often contains a logo or the website name:

**Example**

<!DOCTYPE html>

<html lang="en">

<head>

<title>CSS Website Layout</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

body {

margin: 0;

}

/\* Style the header \*/

.header {

background-color: sky;

padding: 20px;

text-align: center;

}

</style>

</head>

<body>

<div class="header">

<h1>Header</h1>

</div>

</body>

</html>

## 2)Navigation Bar

A navigation bar contains a list of links to help visitors navigating through your website:

**Example**

<!DOCTYPE html>

<html lang="en">

<head>

<title>CSS Website Layout</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

\* {

box-sizing: border-box;

}

body {

margin: 0;

}

/\* Style the header \*/

.header {

background-color: #f1f1f1;

padding: 20px;

text-align: center;

}

/\* Style the top navigation bar \*/

.topnav {

overflow: hidden;

background-color: #333;

}

/\* Style the topnav links \*/

.topnav a {

float: left;

display: block;

color: #f2f2f2;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

/\* Change color on hover \*/

.topnav a:hover {

background-color: #ddd;

color: black;

}

</style>

</head>

<body>

<div class="header">

<h1>Header</h1>

</div>

<div class="topnav">

<a href="#">Link</a>

<a href="#">Link</a>

<a href="#">Link</a>

</div>

</body>

</html>

## 3) Content

The layout in this section, often depends on the target users. The most common layout is one (or combining them) of the following:

* **1-column** (often used for mobile browsers)
* **2-column** (often used for tablets and laptops)
* **3-column layout** (only used for desktops)

We will create a 3-column layout, and change it to a 1-column layout on smaller screens:

**Example**

<!DOCTYPE html>

<html lang="en">

<head>

<title>CSS Website Layout</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

\* {

box-sizing: border-box;

}

body {

margin: 0;

}

/\* Style the header \*/

.header {

background-color: #f1f1f1;

padding: 20px;

text-align: center;

}

/\* Style the top navigation bar \*/

.topnav {

overflow: hidden;

background-color: #333;

}

/\* Style the topnav links \*/

.topnav a {

float: left;

display: block;

color: #f2f2f2;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

/\* Change color on hover \*/

.topnav a:hover {

background-color: #ddd;

color: black;

}

/\* Create three equal columns that floats next to each other \*/

.column {

float: left;

width: 33.33%;

padding: 15px;

}

/\* Clear floats after the columns \*/

.row:after {

content: "";

display: table;

clear: both;

}

/\* Responsive layout - makes the three columns stack on top of each other instead of next to each other \*/

@media screen and (max-width:600px) {

.column {

width: 100%;

}

}

</style>

</head>

<body>

<div class="header">

<h1>Header</h1>

<p>Resize the browser window to see the responsive effect.</p>

</div>

<div class="topnav">

<a href="#">Link</a>

<a href="#">Link</a>

<a href="#">Link</a>

</div>

<div class="row">

<div class="column">

<h2>Column</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

</div>

<div class="column">

<h2>Column</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

</div>

<div class="column">

<h2>Column</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

</div>

</div>

</body>

</html>

### 4)Unequal Columns

The main content is the biggest and the most important part of your site.

It is common with **unequal** column widths, so that most of the space is reserved for the main content. The side content (if any) is often used as an alternative navigation or to specify information relevant to the main content. Change the widths as you like, only remember that it should add up to 100% in total:

**Example**

<!DOCTYPE html>

<html lang="en">

<head>

<title>CSS Website Layout</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

\* {

box-sizing: border-box;

}

body {

margin: 0;

}

/\* Style the header \*/

.header {

background-color: #f1f1f1;

padding: 20px;

text-align: center;

}

/\* Style the top navigation bar \*/

.topnav {

overflow: hidden;

background-color: #333;

}

/\* Style the topnav links \*/

.topnav a {

float: left;

display: block;

color: #f2f2f2;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

/\* Change color on hover \*/

.topnav a:hover {

background-color: #ddd;

color: black;

}

/\* Create three unequal columns that floats next to each other \*/

.column {

float: left;

padding: 10px;

}

/\* Left and right column \*/

.column.side {

width: 25%;

}

/\* Middle column \*/

.column.middle {

width: 50%;

}

/\* Clear floats after the columns \*/

.row:after {

content: "";

display: table;

clear: both;

}

/\* Responsive layout - makes the three columns stack on top of each other instead of next to each other \*/

@media screen and (max-width: 600px) {

.column.side, .column.middle {

width: 100%;

}

}

</style>

</head>

<body>

<div class="header">

<h1>Header</h1>

<p>Resize the browser window to see the responsive effect.</p>

</div>

<div class="topnav">

<a href="#">Link</a>

<a href="#">Link</a>

<a href="#">Link</a>

</div>

<div class="row">

<div class="column side">

<h2>Side</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit..</p>

</div>

<div class="column middle">

<h2>Main Content</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

</div>

<div class="column side">

<h2>Side</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit..</p>

</div>

</div>

</body>

</html>

## 5)Footer

The footer is pelaced at the bottom of your page. It often contains information like copyright and contact info:

### Example

<!DOCTYPE html>

<html lang="en">

<head>

<title>CSS Website Layout</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

\* {

box-sizing: border-box;

}

body {

margin: 0;

}

/\* Style the header \*/

.header {

background-color: #f1f1f1;

padding: 20px;

text-align: center;

}

/\* Style the top navigation bar \*/

.topnav {

overflow: hidden;

background-color: #333;

}

/\* Style the topnav links \*/

.topnav a {

float: left;

display: block;

color: #f2f2f2;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

/\* Change color on hover \*/

.topnav a:hover {

background-color: #ddd;

color: black;

}

/\* Create three unequal columns that floats next to each other \*/

.column {

float: left;

padding: 10px;

}

/\* Left and right column \*/

.column.side {

width: 25%;

}

/\* Middle column \*/

.column.middle {

width: 50%;

}

/\* Clear floats after the columns \*/

.row:after {

content: "";

display: table;

clear: both;

}

/\* Responsive layout - makes the three columns stack on top of each other instead of next to each other \*/

@media screen and (max-width: 600px) {

.column.side, .column.middle {

width: 100%;

}

}

/\* Style the footer \*/

.footer {

background-color: #f1f1f1;

padding: 10px;

text-align: center;

}

</style>

</head>

<body>

<div class="header">

<h1>Header</h1>

<p>Resize the browser window to see the responsive effect.</p>

</div>

<div class="topnav">

<a href="#">Link</a>

<a href="#">Link</a>

<a href="#">Link</a>

</div>

<div class="row">

<div class="column side">

<h2>Side</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit..</p>

</div>

<div class="column middle">

<h2>Main Content</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit amet pretium urna. Vivamus venenatis velit nec neque ultricies, eget elementum magna tristique. Quisque vehicula, risus eget aliquam placerat, purus leo tincidunt eros, eget luctus quam orci in velit. Praesent scelerisque tortor sed accumsan convallis.</p>

</div>

<div class="column side">

<h2>Side</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit..</p>

</div>

</div>

<div class="footer">

<p>Footer</p>

</div>

</body>

</html>

**Thank you**