MODULE -2

STYLING WITH CSS

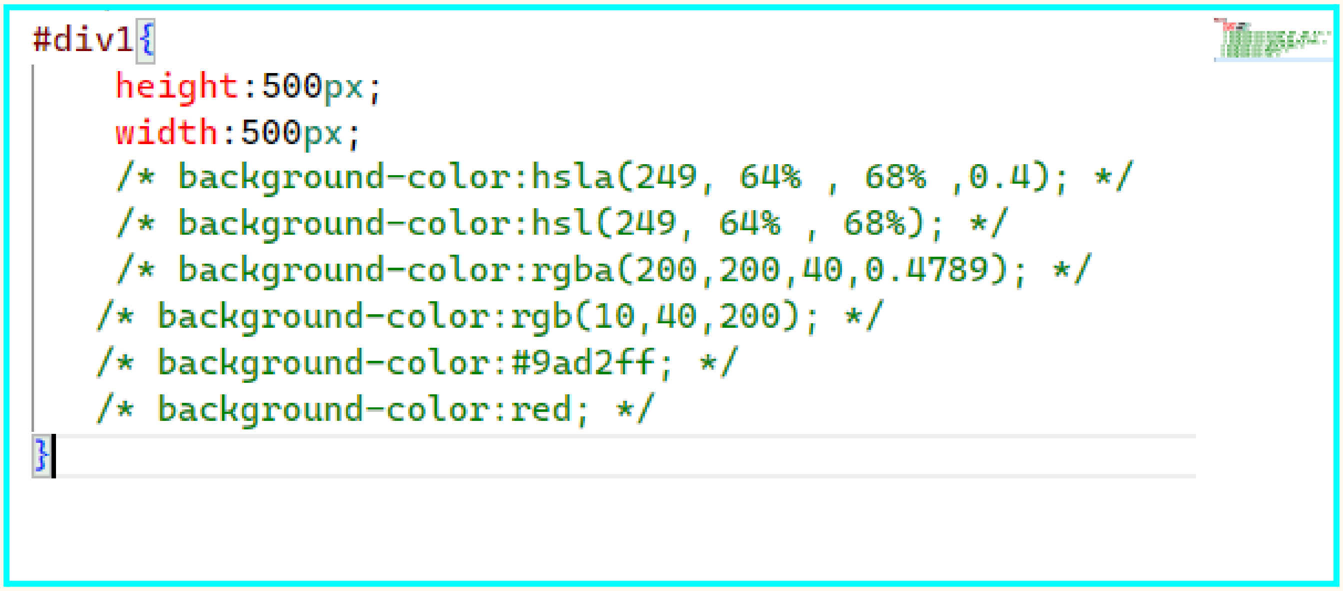
-BY

Chirag Maini

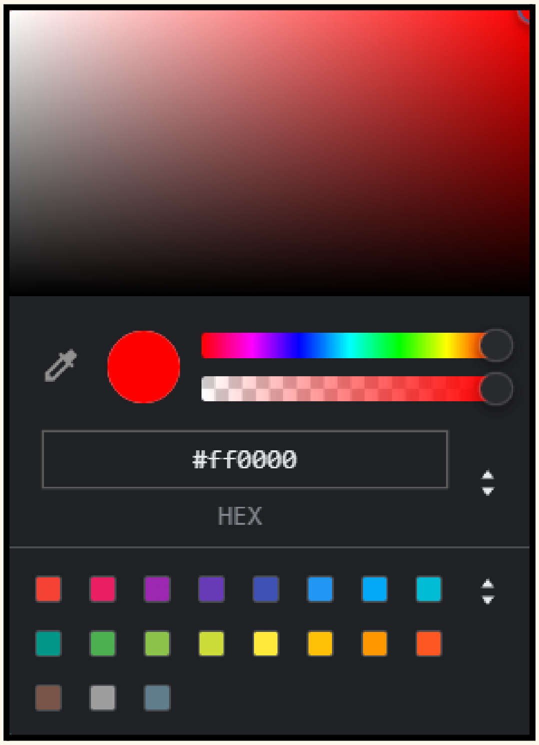
NSUT’25(EE)

->>>> Adding Colors: <<<<-:

1. Name
2. Hexadecimal value
3. rgb
4. rgba
5. Hsl
6. hsla

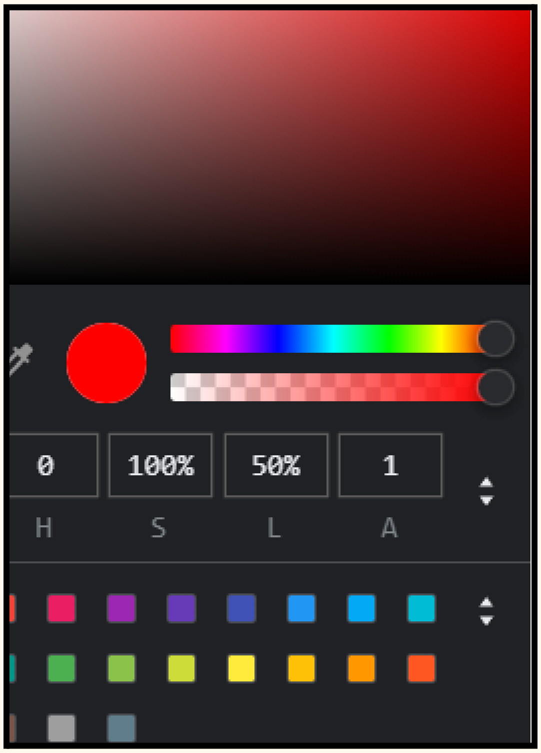


```
#div1{
    height:500px;
    width:500px;
    /* background-color:hsla(249, 64% , 68% ,0.4); */
    /* background-color:hsl(249, 64% , 68%); */
    /* background-color:rgba(200,200,40,0.4789); */
   /* background-color:rgb(10,40,200); */
   /* background-color:#9ad2ff; */
   /* background-color:red; */
}
```

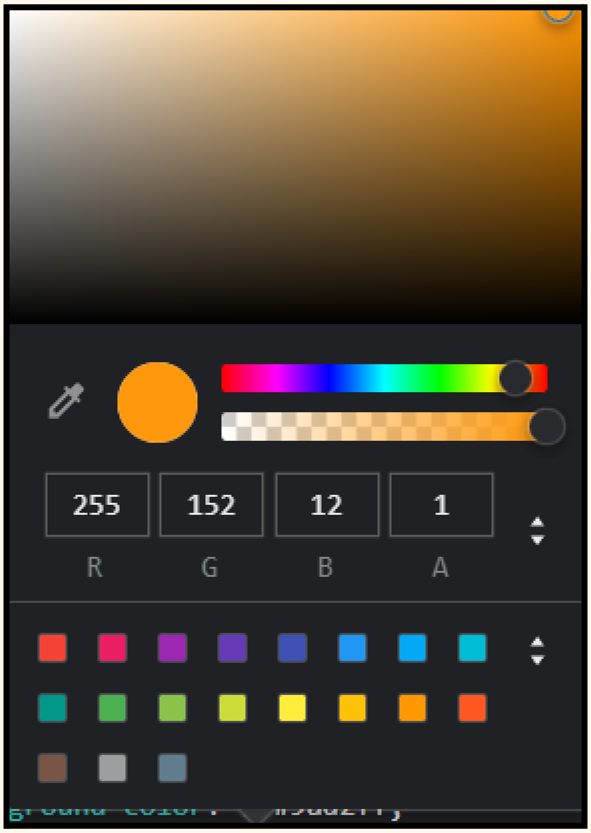


#ff0000

HEX



0	100%	50%	1
H	S	L	A



255	152	12	1
R	G	B	A

->>>> CSS Units <<<<-

There are two types of units in CSS :-

1. Absolute units(cm , inch , pixels(px))
2. Relative units (% {relative to the parent} , vh{view port height} ,vw{view port width} , rem {relative to font-size of the root element ie HTML}, em{relative to the font-size of the parent element}

Key Points :

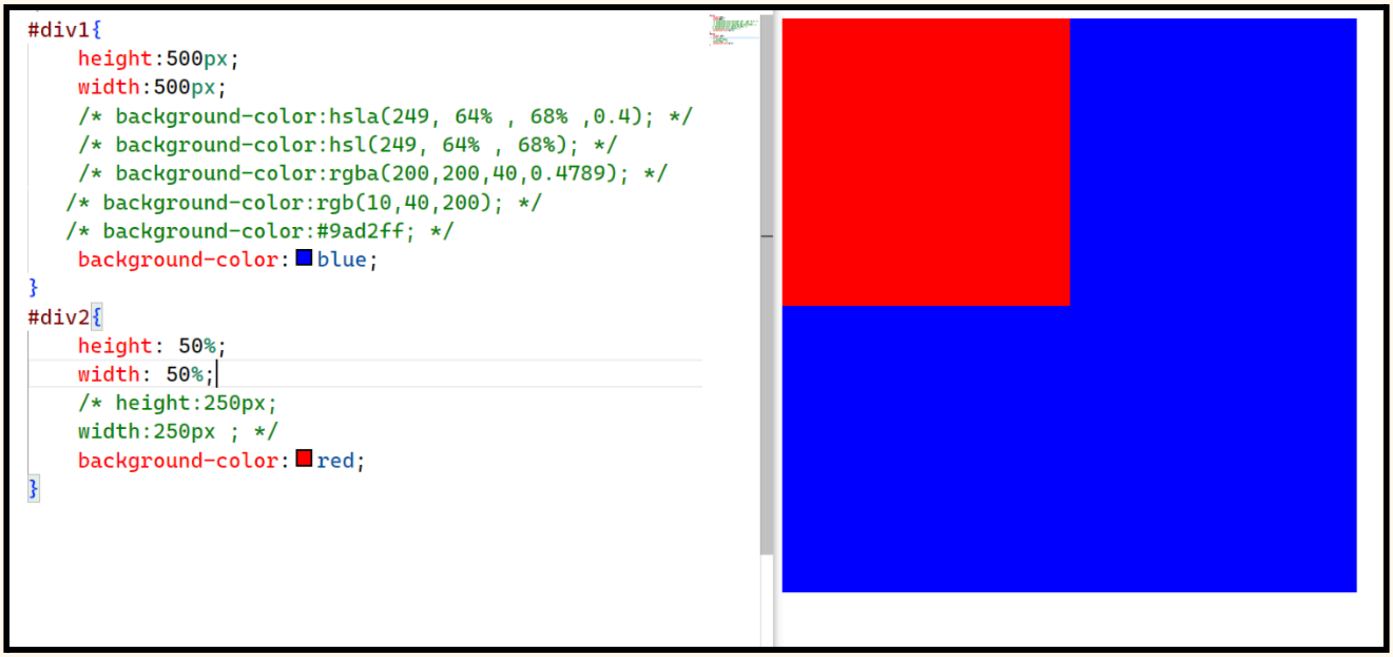
-> css pixel is the smallest unit (1css px= 1/96 inch)

-> 1 CSS pixel may or may not be equal to the 1 device pixel

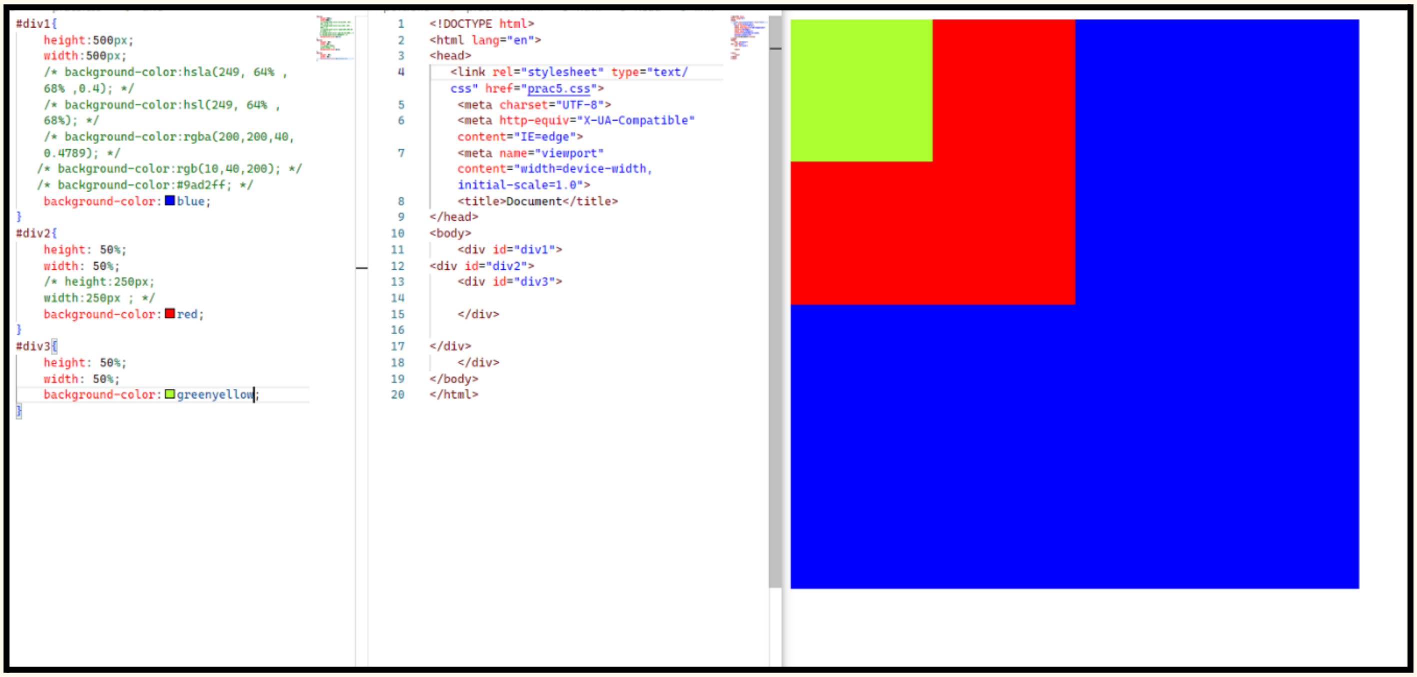
->View Port(vw) is basically the visible area of the browser

So vh is the visible height of the browser and vw is the visible width of the browser.

-> rem is the unit which is relative to the font-size of the root element ie html element (in px).

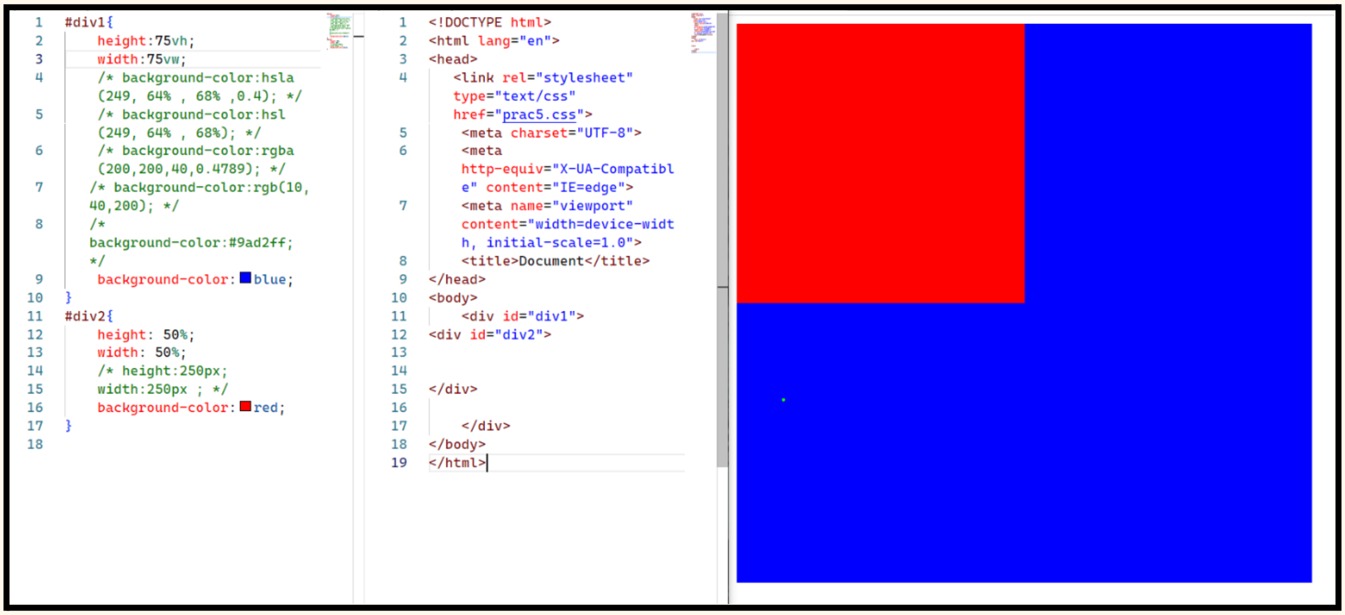


```css
#div1{
    height:500px;
    width:500px;
    /* background-color:hsla(249, 64% , 68% ,0.4); */
    /* background-color:hsl(249, 64% , 68%); */
    /* background-color:rgba(200,200,40,0.4789); */
   /* background-color:rgb(10,40,200); */
   /* background-color:#9ad2ff; */
    background-color: blue;
}
#div2{
    height: 50%;
    width: 50%;
    /* height:250px;
    width:250px ; */
    background-color: red;
}
```



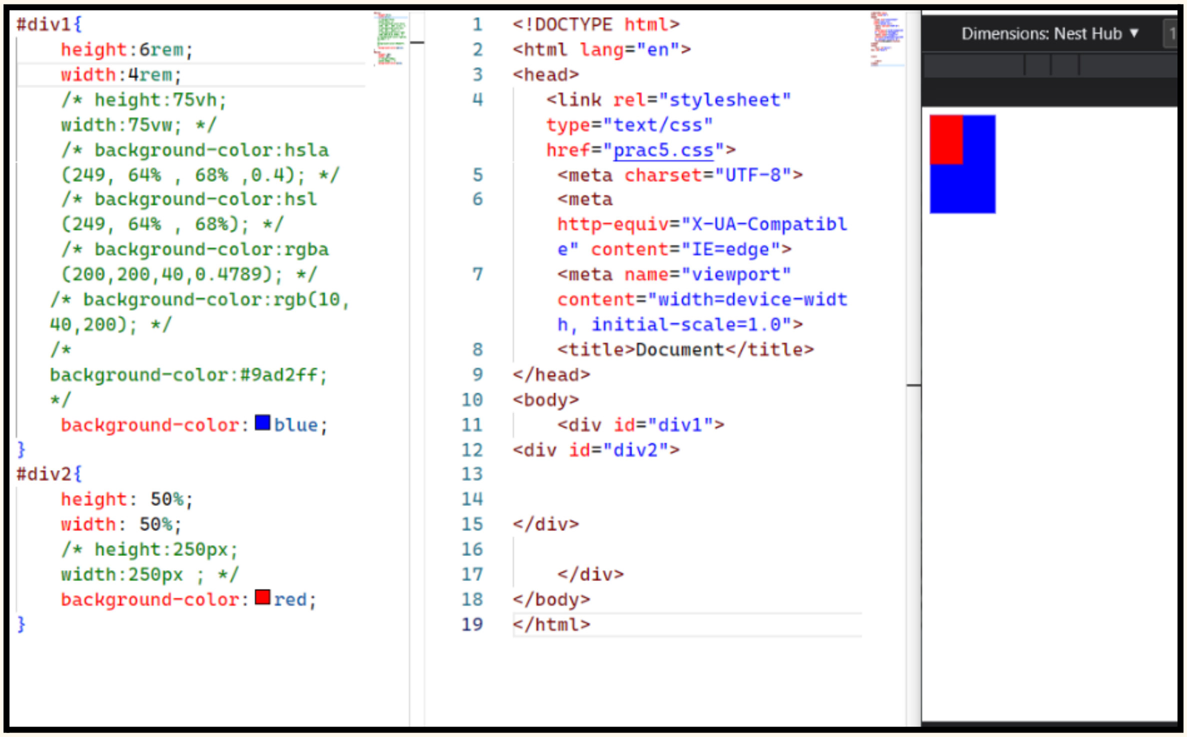
```css
#div1{
    height:500px;
    width:500px;
    /* background-color:hsla(249, 64% ,
    68% ,0.4); */
    /* background-color:hsl(249, 64% ,
    68%); */
    /* background-color:rgba(200,200,40,
    0.4789); */
   /* background-color:rgb(10,40,200); */
   /* background-color:#9ad2ff; */
    background-color:blue;
}
#div2{
    height: 50%;
    width: 50%;
    /* height:250px;
    width:250px ; */
    background-color:red;
}
#div3{
    height: 50%;
    width: 50%;
    background-color:greenyellow;
}
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
   <link rel="stylesheet" type="text/
   css" href="prac5.css">
    <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>
</head>
<body>
    <div id="div1">
<div id="div2">
    <div id="div3">

    </div>

</div>
    </div>
</body>
</html>
```



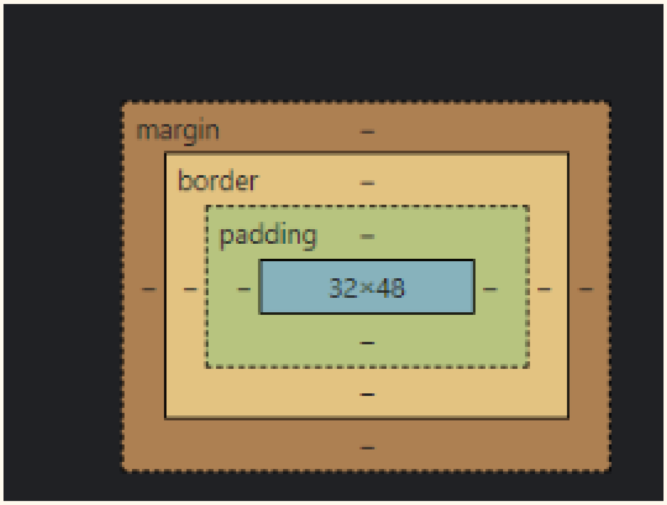
```css
#div1{
    height:75vh;
    width:75vw;
    /* background-color:hsla
    (249, 64% , 68% ,0.4); */
    /* background-color:hsl
    (249, 64% , 68%); */
    /* background-color:rgba
    (200,200,40,0.4789); */
   /* background-color:rgb(10,
   40,200); */
   /*
   background-color:#9ad2ff;
   */
    background-color: 🟦blue;
}
#div2{
    height: 50%;
    width: 50%;
    /* height:250px;
    width:250px ; */
    background-color: 🟥red;
}
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
   <link rel="stylesheet"
   type="text/css"
   href="prac5.css">
    <meta charset="UTF-8">
    <meta
    http-equiv="X-UA-Compatibl
    e" content="IE=edge">
    <meta name="viewport"
    content="width=device-widt
    h, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <div id="div1">
<div id="div2">


</div>

    </div>
</body>
</html>
```



```css
#div1{
    height:6rem;
    width:4rem;
    /* height:75vh;
    width:75vw; */
    /* background-color:hsla
    (249, 64% , 68% ,0.4); */
    /* background-color:hsl
    (249, 64% , 68%); */
    /* background-color:rgba
    (200,200,40,0.4789); */
   /* background-color:rgb(10,
   40,200); */
   /*
   background-color:#9ad2ff;
   */
    background-color: 🟦blue;
}
#div2{
    height: 50%;
    width: 50%;
    /* height:250px;
    width:250px ; */
    background-color: 🟥red;
}
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
   <link rel="stylesheet"
   type="text/css"
   href="prac5.css">
    <meta charset="UTF-8">
    <meta
    http-equiv="X-UA-Compatibl
    e" content="IE=edge">
    <meta name="viewport"
    content="width=device-widt
    h, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <div id="div1">
<div id="div2">


</div>

    </div>
</body>
</html>
```



How to add border to an element:-

Syntax:

Background-width:4px;

Background-style:solid;

background-color: red;

OR

background : 4px solid red;

IMP POINT:

-> Background-width:4px;

# 2px is the width of all borders.

-> background-width : 2px 4px;

# 2px is the width of the top and bottom border.

# 4px is the width of left and right.

-> background-width : 2px 4px 6px;

# 2px is the width of the top border.

# 6px is the width of the bottom border.

# 4px is the width of left and right.

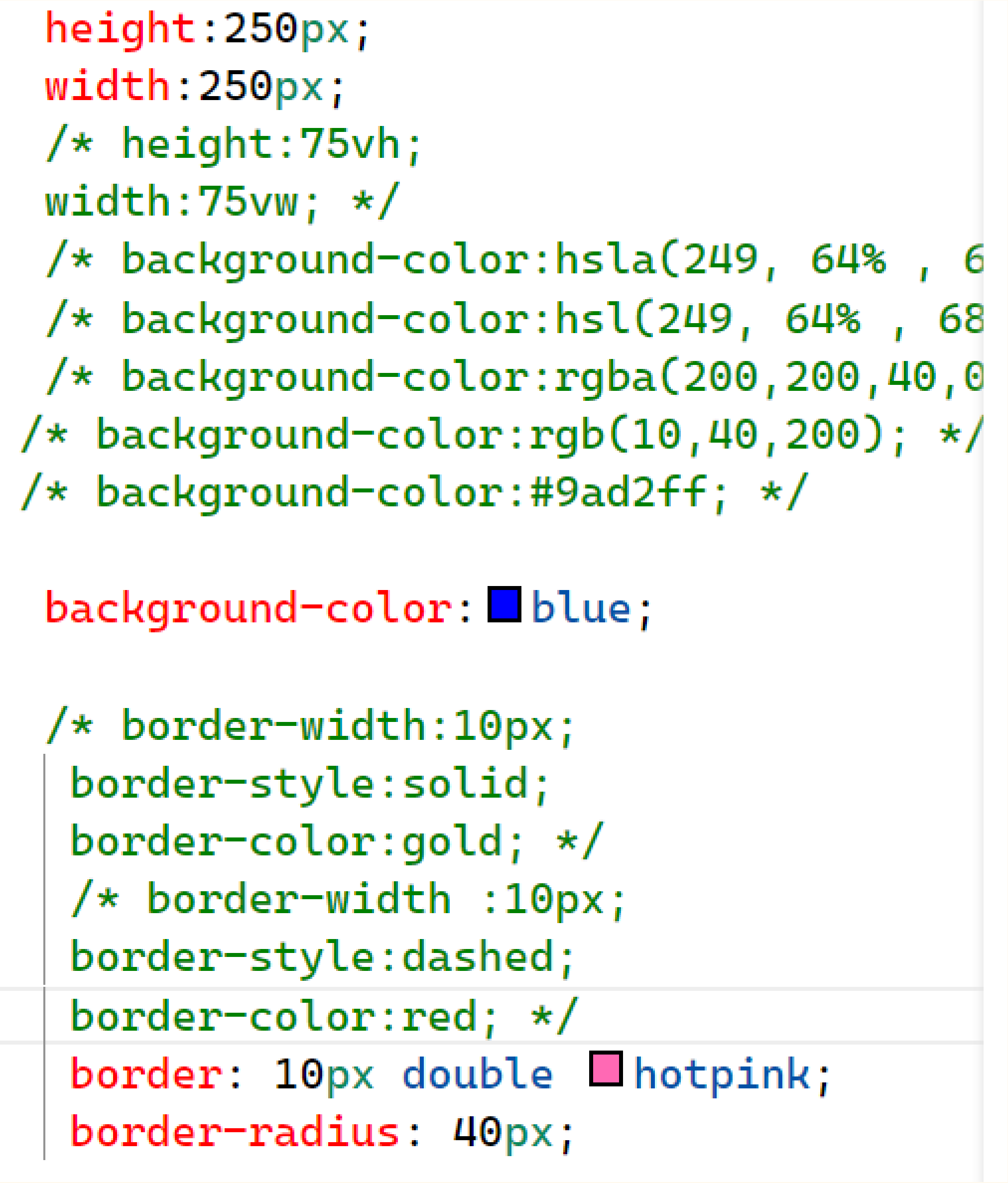
-> background-width : 2px 4px 6px 8px ;

# 2px is the width of the top border.

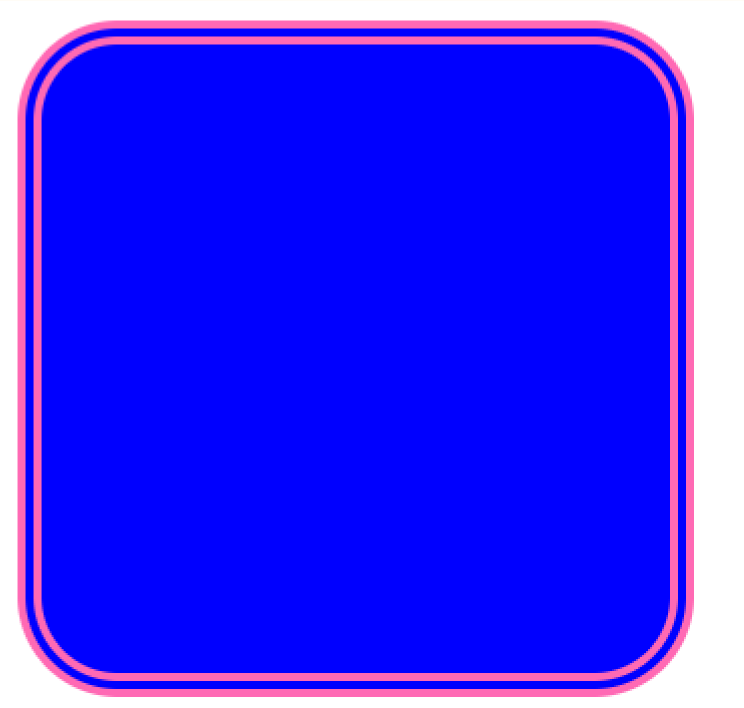
# 4px is the width of the right border.

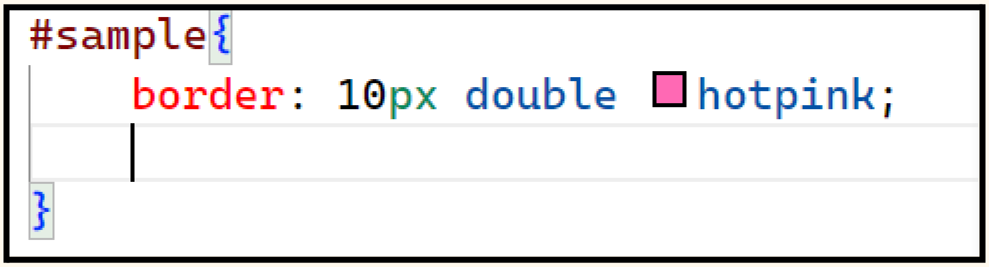
# 6px is the width of the bottom border.

# 8px is the width of the left border.

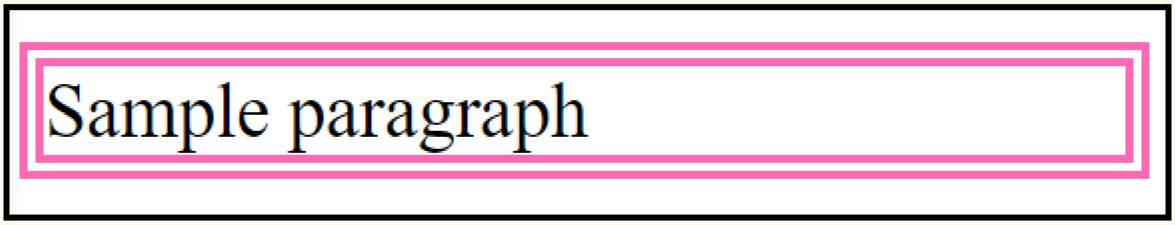


```
  height:250px;
  width:250px;
  /* height:75vh;
  width:75vw; */
  /* background-color:hsla(249, 64% , 6
  /* background-color:hsl(249, 64% , 68
  /* background-color:rgba(200,200,40,0
 /* background-color:rgb(10,40,200); */
 /* background-color:#9ad2ff; */

  background-color:blue;

  /* border-width:10px;
   border-style:solid;
   border-color:gold; */
   /* border-width :10px;
   border-style:dashed;
   border-color:red; */
   border: 10px double hotpink;
   border-radius: 40px;
```

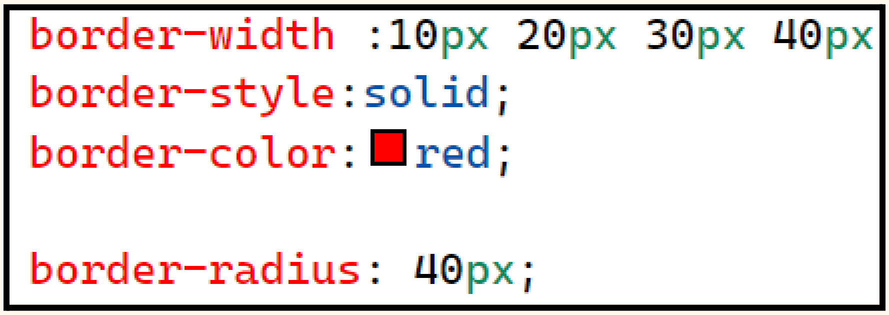




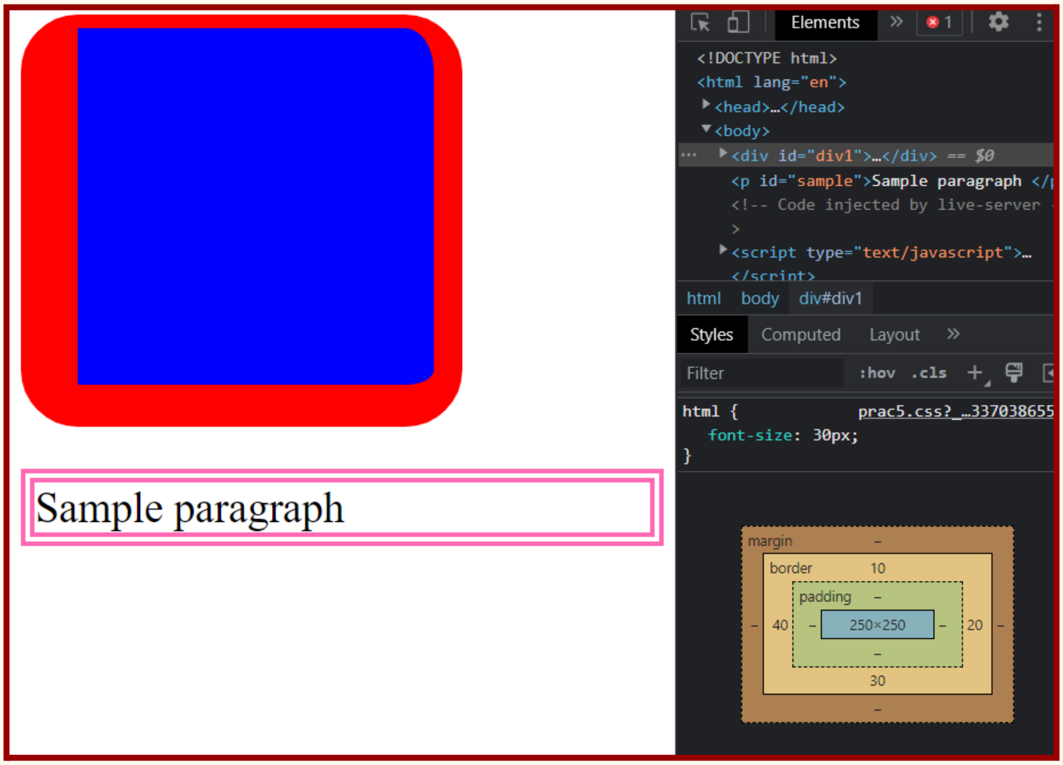
```
#sample{
        border: 10px double hotpink;
        
}
```



Sample paragraph



```
border-width :10px 20px 30px 40px
border-style:solid;
border-color:red;

border-radius: 40px;
```

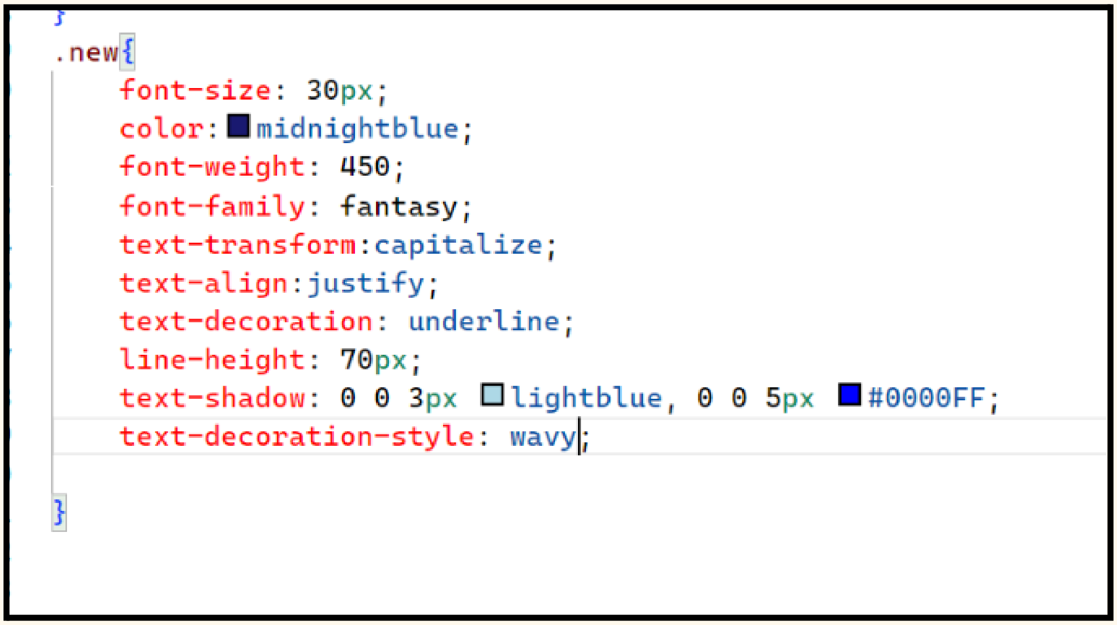


Sample paragraph

```
<!DOCTYPE html>
<html lang="en">
  ▶<head>…</head>
  ▼<body>
…   ▶<div id="div1">…</div> == $0
      <p id="sample">Sample paragraph </p
      <!-- Code injected by live-server -
      >
    ▶<script type="text/javascript">…
      </script>
```

```
html {                    prac5.css?_…337038655
   font-size: 30px;
}
```

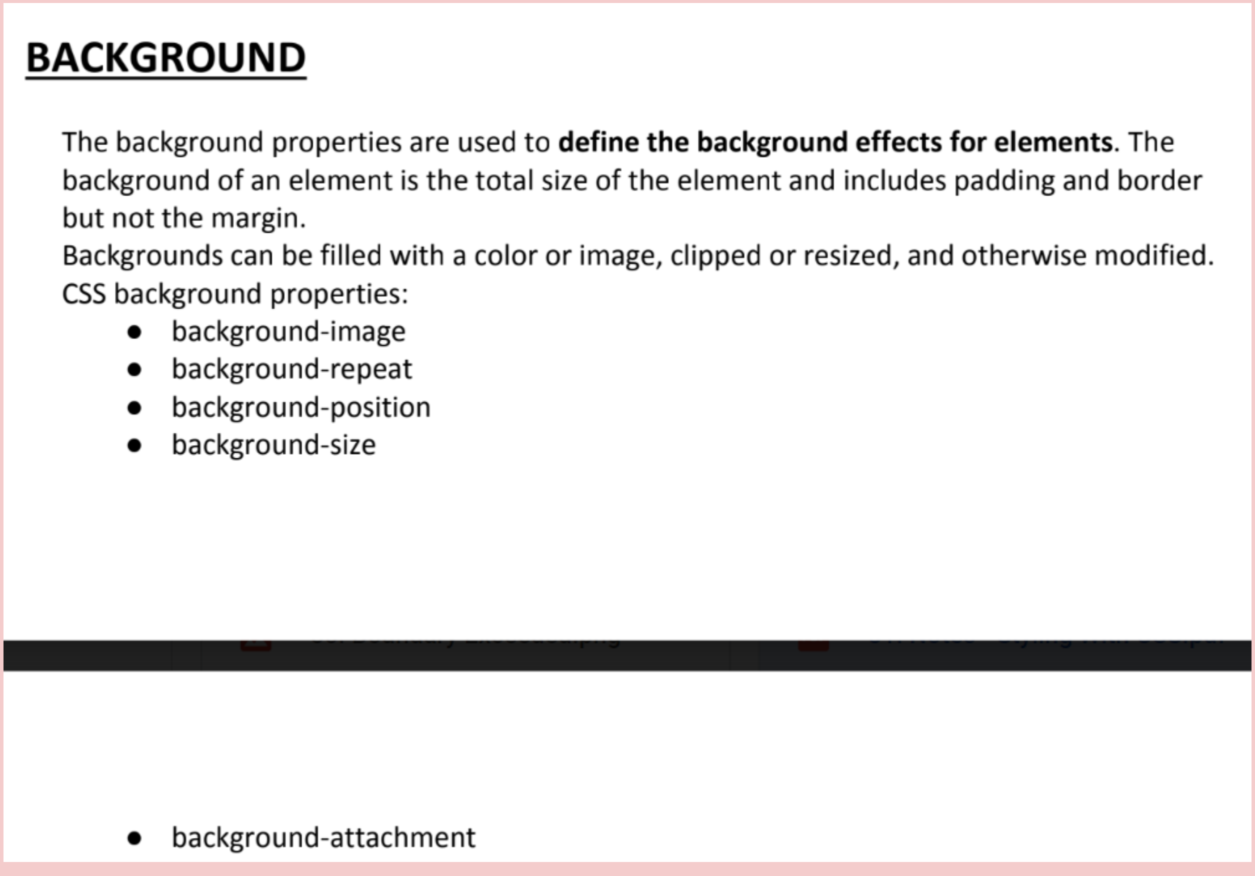
Text Styling:



```
}
.new{
    font-size: 30px;
    color: ■midnightblue;
    font-weight: 450;
    font-family: fantasy;
    text-transform:capitalize;
    text-align:justify;
    text-decoration: underline;
    line-height: 70px;
    text-shadow: 0 0 3px □lightblue, 0 0 5px ■#0000FF;
    text-decoration-style: wavy;

}
```

Cascading Style Sheets (CSS) Is A Style Sheet Language Used For Describing The Presentation Of A Document Written In A Markup Language Such As HTML Or XML (Including XML Dialects Such As SVG, MathML Or XHTML).[1] CSS Is A Cornerstone Technology Of The World Wide Web, Alongside HTML And JavaScript.CSS Is Designed To Enable The Separation Of Presentation And Content, Including Layout, Colors, And Fonts.[3] This Separation Can Improve Content Accessibility; Provide More Flexibility And Control In The Specification Of Presentation Characteristics; Enable Multiple Web Pages To Share Formatting By Specifying The Relevant CSS In A Separate .Css File, Which Reduces Complexity And Repetition In The Structural Content; And Enable The .Css File To Be Cached To Improve The Page Load Speed Between The Pages That Share The File And Its Formatting

->>>> Background <<<-:



BACKGROUND

The background properties are used to **define the background effects for elements**. The background of an element is the total size of the element and includes padding and border but not the margin.
Backgrounds can be filled with a color or image, clipped or resized, and otherwise modified.
CSS background properties:

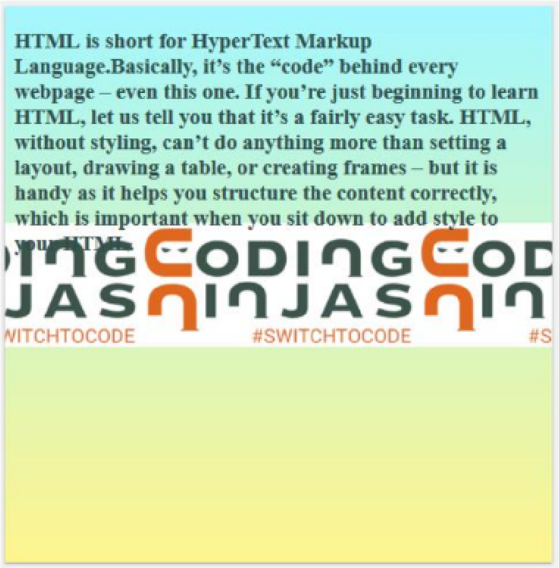
- background-image
- background-repeat
- background-position
- background-size

- background-attachment

Eg., the below CSS code when applied to a web page

```
body {
    background-image: url("https://blog.codingninjas.in/wp-
     content/uploads/2017/01/cropped-Final_logo_switchtocode-01.png"), linear-
     gradient(#a3f7ff, #fff58e);
    background-repeat: repeat-x;
    background-position: center center;
    background-attachment: fixed;
}
```

will show the web page like this:



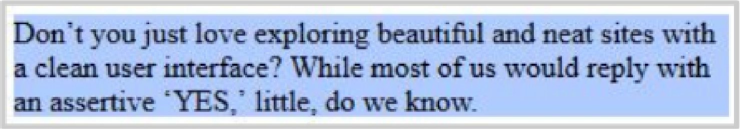
HTML is short for HyperText Markup Language.Basically, it's the "code" behind every webpage – even this one. If you're just beginning to learn HTML, let us tell you that it's a fairly easy task. HTML, without styling, can't do anything more than setting a layout, drawing a table, or creating frames – but it is handy as it helps you structure the content correctly, which is important when you sit down to add style to your HTML.

background-color

The **background-color** property sets the **background color of an element**. It has the same value as that of the color property.

Eg., `<p style="background-color: #afcbff;"> Don't you just love exploring beautiful and neat sites with a clean user interface? While most of us would reply with an assertive 'YES,' little, do we know. </p>`

will show like:



Don't you just love exploring beautiful and neat sites with a clean user interface? While most of us would reply with an assertive 'YES,' little, do we know.

5.2. background-image

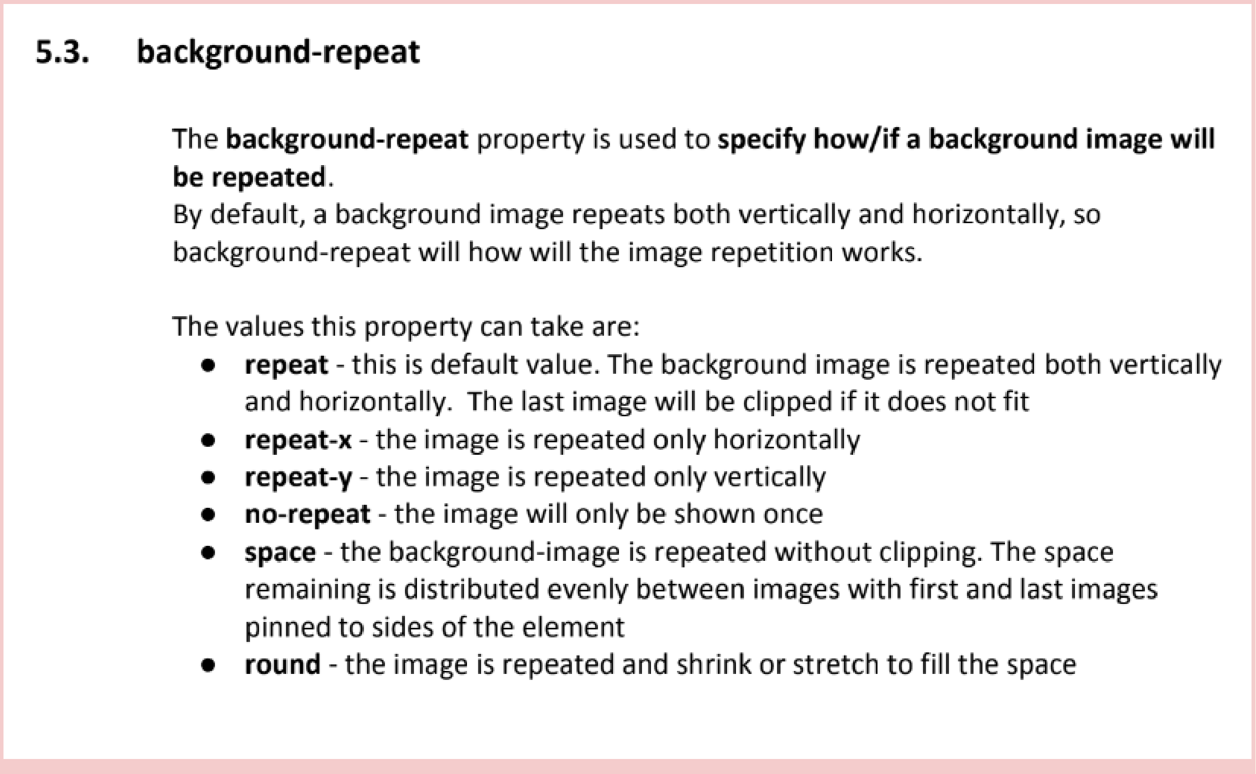
The **background-image** property is used to **specify an image to use as the background of an element**.
This can set one or more background images for an element.

By default, a background-image is placed at the top-left corner of an element, and is repeated so it covers the entire element both vertically and horizontally.

The values it can take are:

- **url('*URL*')** - specifies the URL of the image. You can specify more than one image by separating the URLs with a comma
- **none** - this is **default** value. No background image will be displayed
- **linear-gradient()** - sets a linear gradient as the background image. At least two colors needed to be mentioned (default direction is top to bottom)
- **radial-gradient()** - sets a radial gradient as the background image. At least two colors needed to be mentioned (default is from center to edges)
- **repeating-linear-gradient()** - repeats a linear gradient
- **repeating-radial-gradient()** - repeats a radial gradient

In the above example, we are using both the image and linear-gradient together.
Try to swap the sequence of image and gradient and see what happens.



5.3. background-repeat

The **background-repeat** property is used to **specify how/if a background image will be repeated**.
By default, a background image repeats both vertically and horizontally, so background-repeat will how will the image repetition works.

The values this property can take are:

- **repeat** - this is default value. The background image is repeated both vertically and horizontally. The last image will be clipped if it does not fit
- **repeat-x** - the image is repeated only horizontally
- **repeat-y** - the image is repeated only vertically
- **no-repeat** - the image will only be shown once
- **space** - the background-image is repeated without clipping. The space remaining is distributed evenly between images with first and last images pinned to sides of the element
- **round** - the image is repeated and shrink or stretch to fill the space

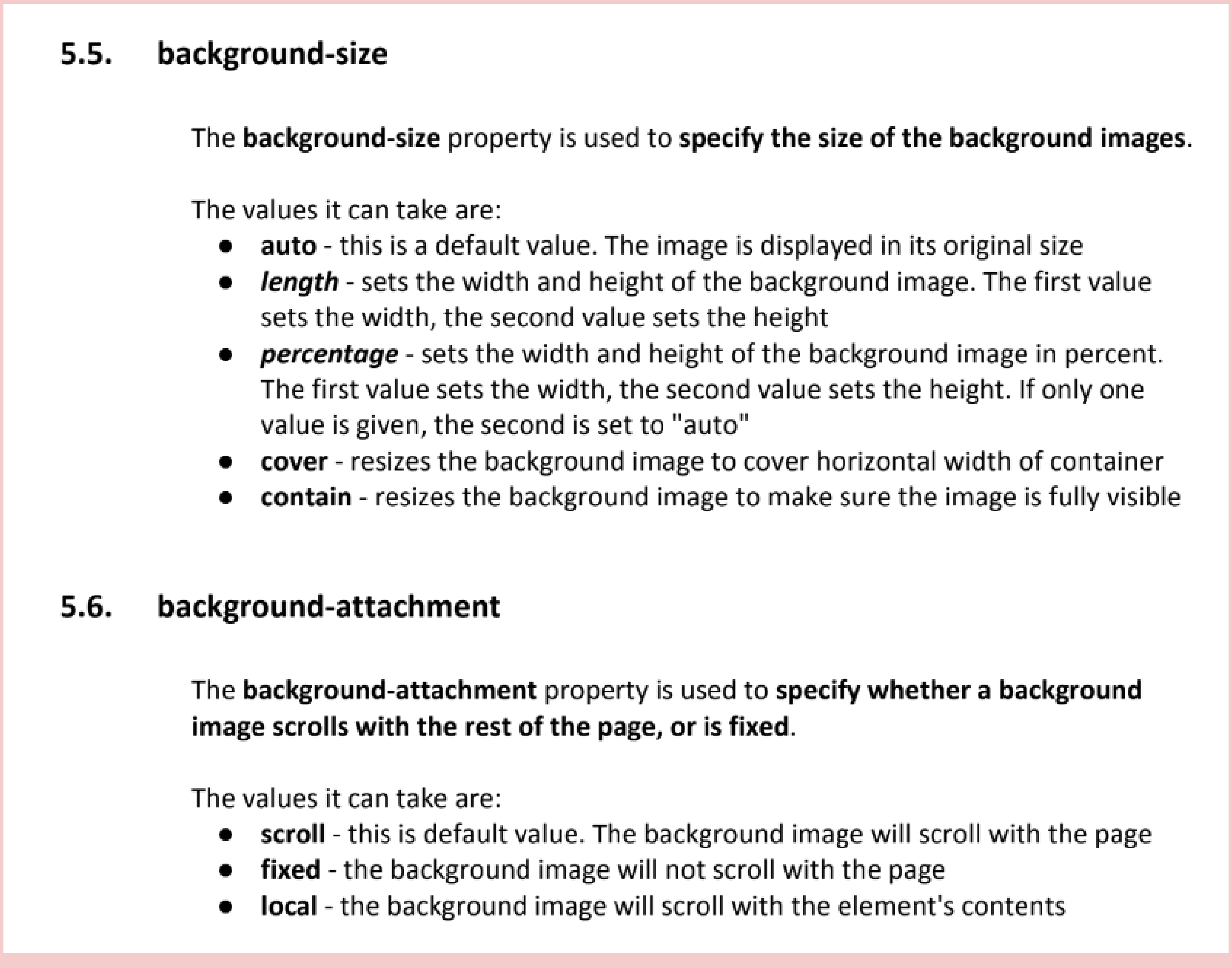
5.4. background-position

The **background-position** property is used to **specify the initial position of a background image**.

By default, a background image is placed at the top-left corner of an element and you can change the position with background-position property.

The values this property can take are(***X represents horizontal position and Y represents vertical position***):

- **X Y** - they both can each take value from one of the following - ***left***, ***right***, ***top***, ***bottom***, ***center***. If one value is specified, the other value will be "center"
- **Xpos Ypos** - specifies the horizontal and vertical position relative to the viewport. Units can be any of the CSS units. If you only specify one value, the other value will be 50%.



5.5. background-size

The **background-size** property is used to **specify the size of the background images**.

The values it can take are:

- **auto** - this is a default value. The image is displayed in its original size
- ***length*** - sets the width and height of the background image. The first value sets the width, the second value sets the height
- ***percentage*** - sets the width and height of the background image in percent. The first value sets the width, the second value sets the height. If only one value is given, the second is set to "auto"
- **cover** - resizes the background image to cover horizontal width of container
- **contain** - resizes the background image to make sure the image is fully visible

5.6. background-attachment

The **background-attachment** property is used to **specify whether a background image scrolls with the rest of the page, or is fixed**.

The values it can take are:

- **scroll** - this is default value. The background image will scroll with the page
- **fixed** - the background image will not scroll with the page
- **local** - the background image will scroll with the element's contents

->>>>> Margins <<<<-

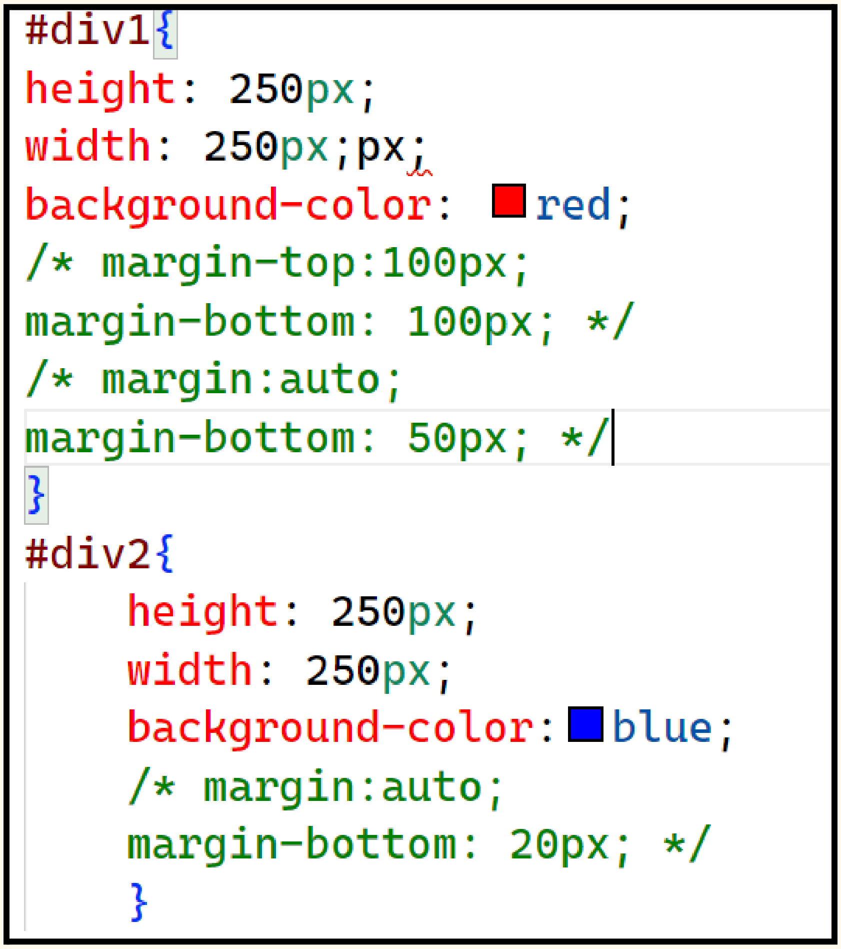
Margins are used to create space around elements, outside of any defined borders.

Properties of Margin:

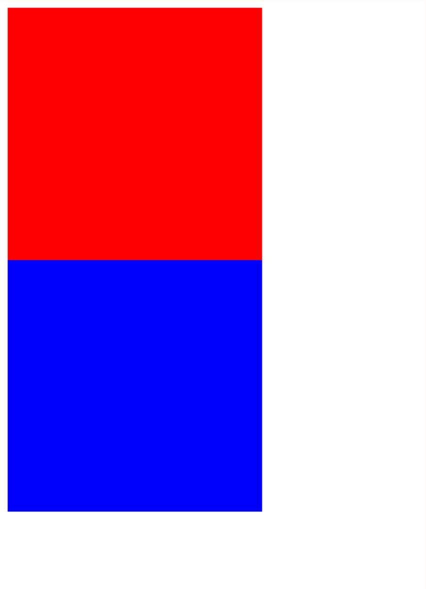
- `margin-top`
- `margin-right`

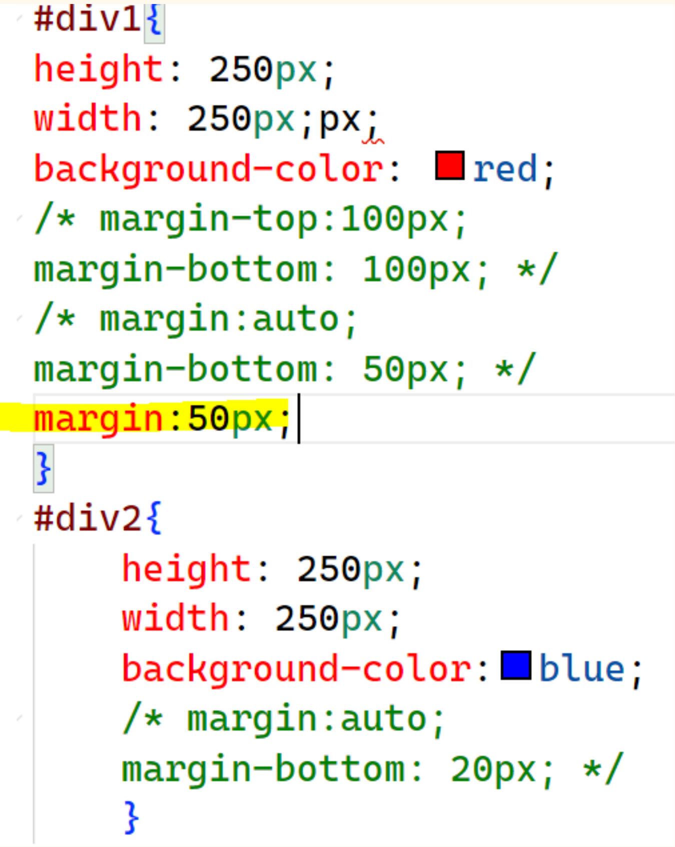
- margin-bottom
- Margin-left

- auto - the browser calculates the margin
- *length* - specifies a margin in px, pt, cm, etc.
- *%* - specifies a margin in % of the width of the containing element
- inherit - specifies that the margin should be inherited from the parent element

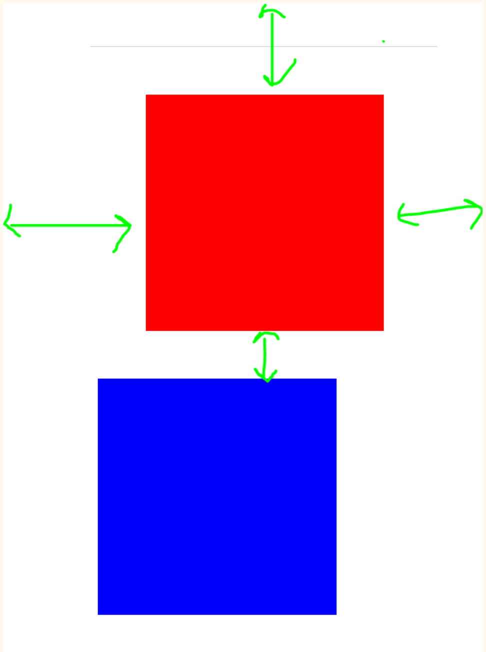


```
#div1{
height: 250px;
width: 250px;px;
background-color: 🟥red;
/* margin-top:100px;
margin-bottom: 100px; */
/* margin:auto;
margin-bottom: 50px; */
}
#div2{
    height: 250px;
    width: 250px;
    background-color:🟦blue;
    /* margin:auto;
    margin-bottom: 20px; */
    }
```





```
#div1{
height: 250px;
width: 250px;px;
background-color: 🟥red;
/* margin-top:100px;
margin-bottom: 100px; */
/* margin:auto;
margin-bottom: 50px; */
margin:50px;
}
#div2{
    height: 250px;
    width: 250px;
    background-color:🟦blue;
    /* margin:auto;
    margin-bottom: 20px; */
    }
```



Few Example :

Margin: 20px;

20 px from each border

Margin: 20px 40px;

Top and bottom :20px;

Left and right :40px

Margin: 20px 40px 60px;

Top:20px

Bottom:60px;

Left and right :40px

Margin: 20px 40px 60px 80px;

Top:20px

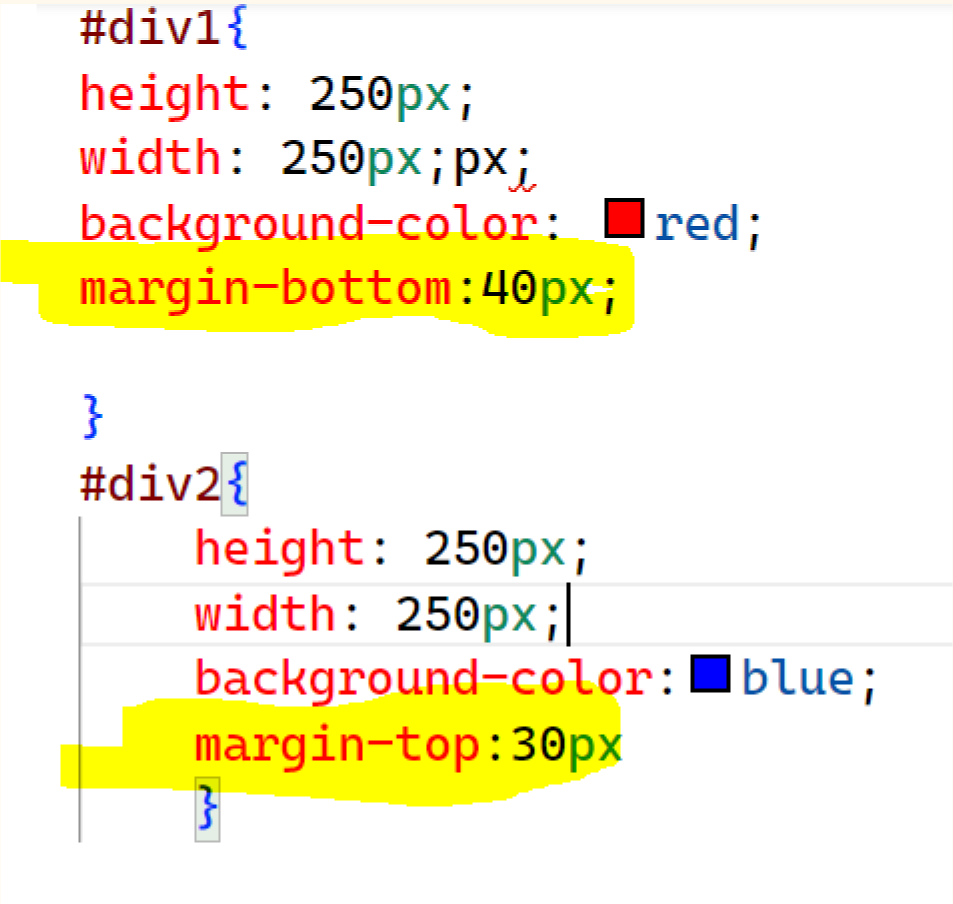
Right:40px

Bottom:60px

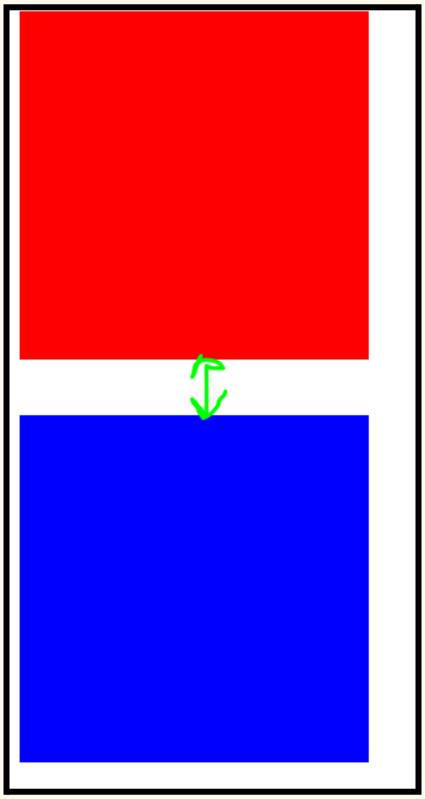
Left:80px

*****NOTE*****

margin:auto; will set the image at the center;



```
#div1{
height: 250px;
width: 250px;px;
background-color: red;
margin-bottom:40px;

}
#div2{
    height: 250px;
    width: 250px;
    background-color: blue;
    margin-top:30px
    }
```



Imp things is that size b/w two blocks would be 40px not 70px it seems that the size of top and bottom should be added but the actual size b/w the blocks would be the largest among the two (in this case 40px as (40>30)

->>>> Padding<<<<-

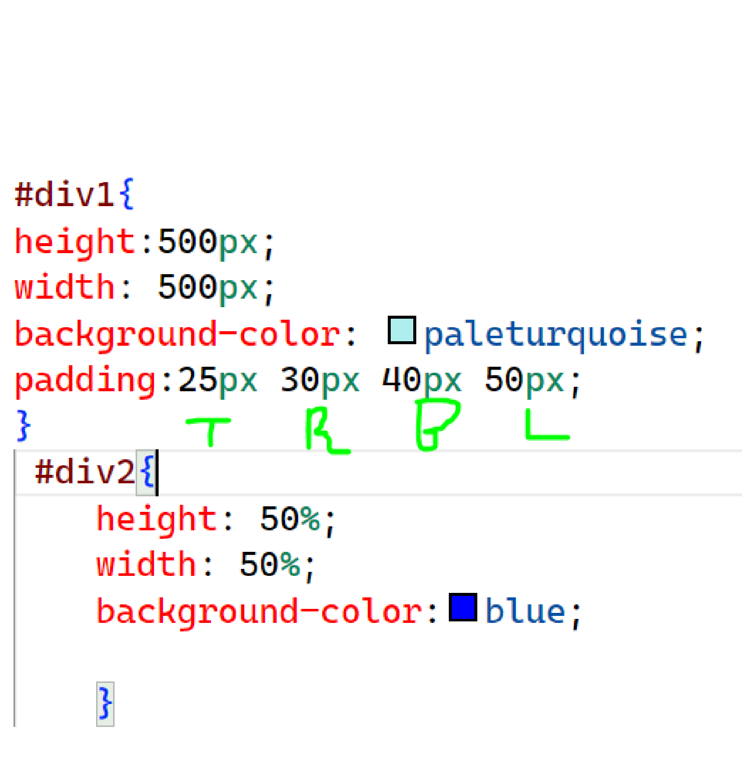
Padding is used to create space around an element's content, inside of any defined borders.

- `padding-top`
- `padding-right`

- `padding-bottom`
- `Padding-left`

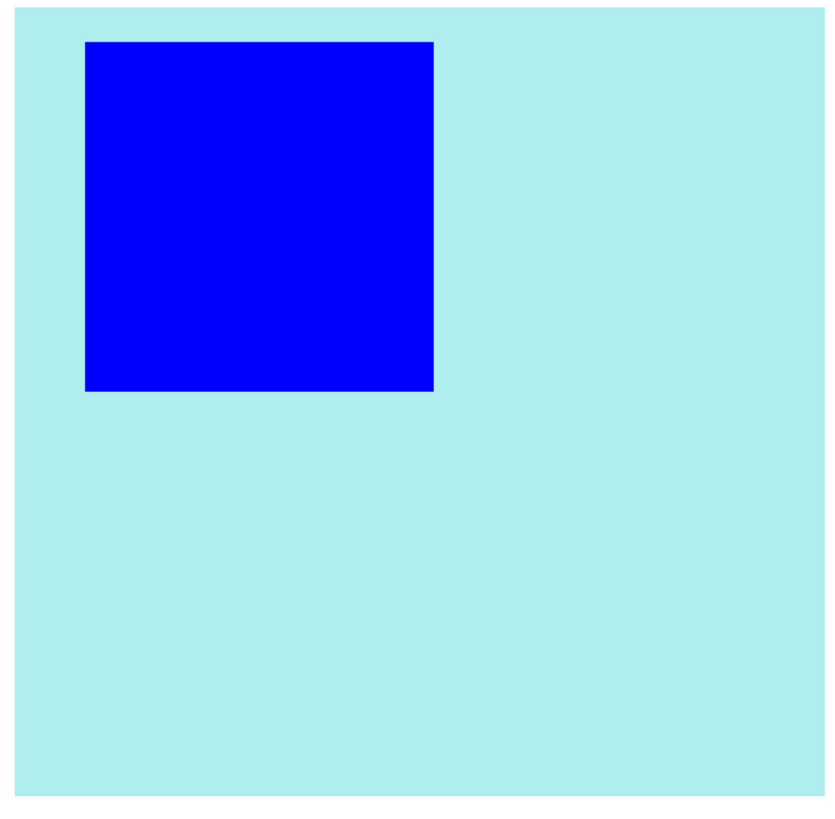
All the padding properties can have the following values:

- *length* - specifies a padding in px, pt, cm, etc.
- *%* - specifies a padding in % of the width of the containing element
- inherit - specifies that the padding should be inherited from the parent element



```
#div1{
height:500px;
width: 500px;
background-color: □paleturquoise;
padding:25px 30px 40px 50px;
}
 #div2{
    height: 50%;
    width: 50%;
    background-color:■blue;

    }
```

T R B L

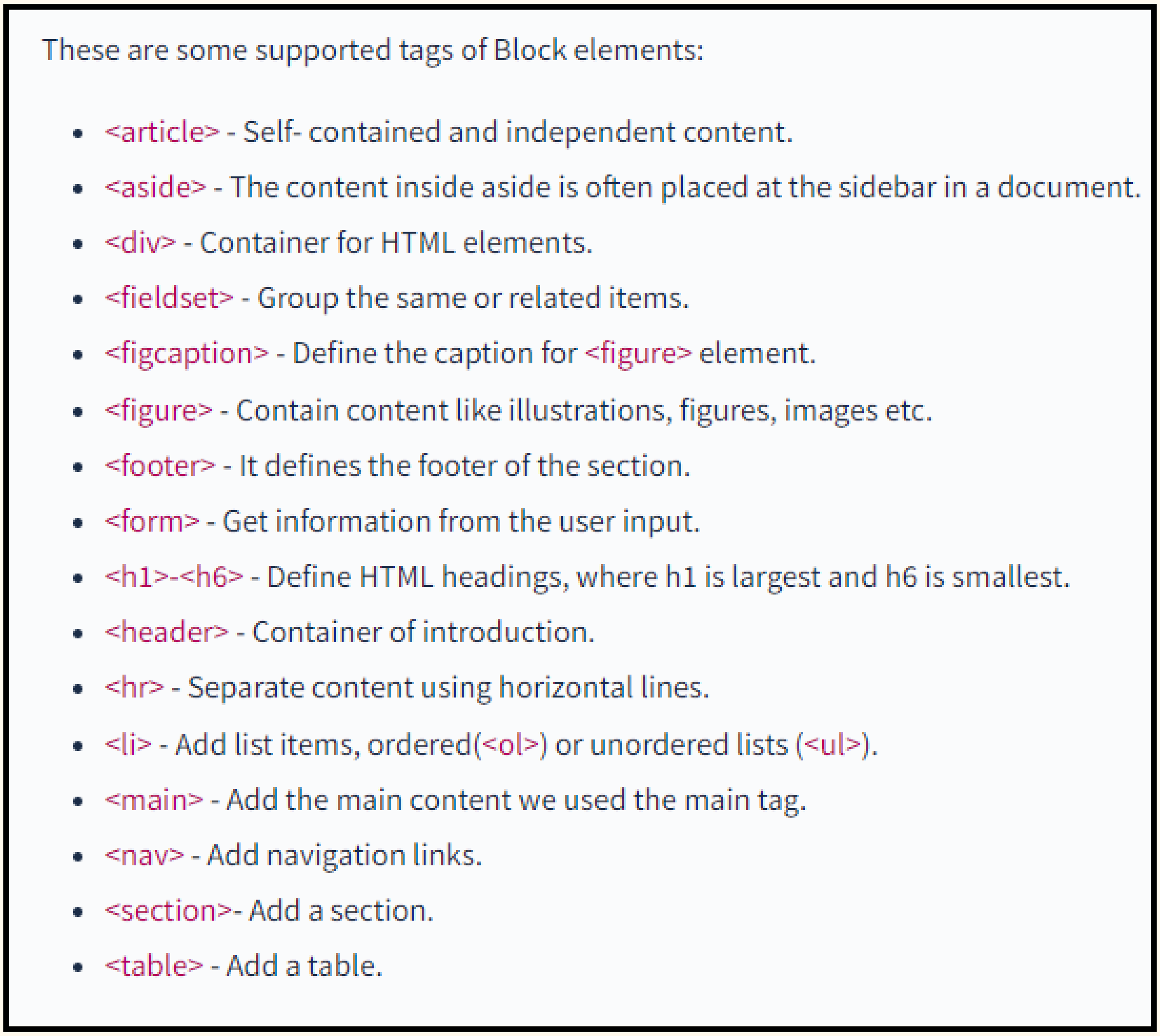


->>>> Display property <<<<-

There are two display values - block and inline elements in HTML.

BLOCK ELEMENT:-

Block elements in HTML begin on a new line and occupy the complete horizontal space of its parent element and have the same height as the content.Block elements begin from a new line by default and cover space to its left and right as far as it can go. The height that it covers is equal to the content height. Also, it covers the whole horizontal space of its parent element.

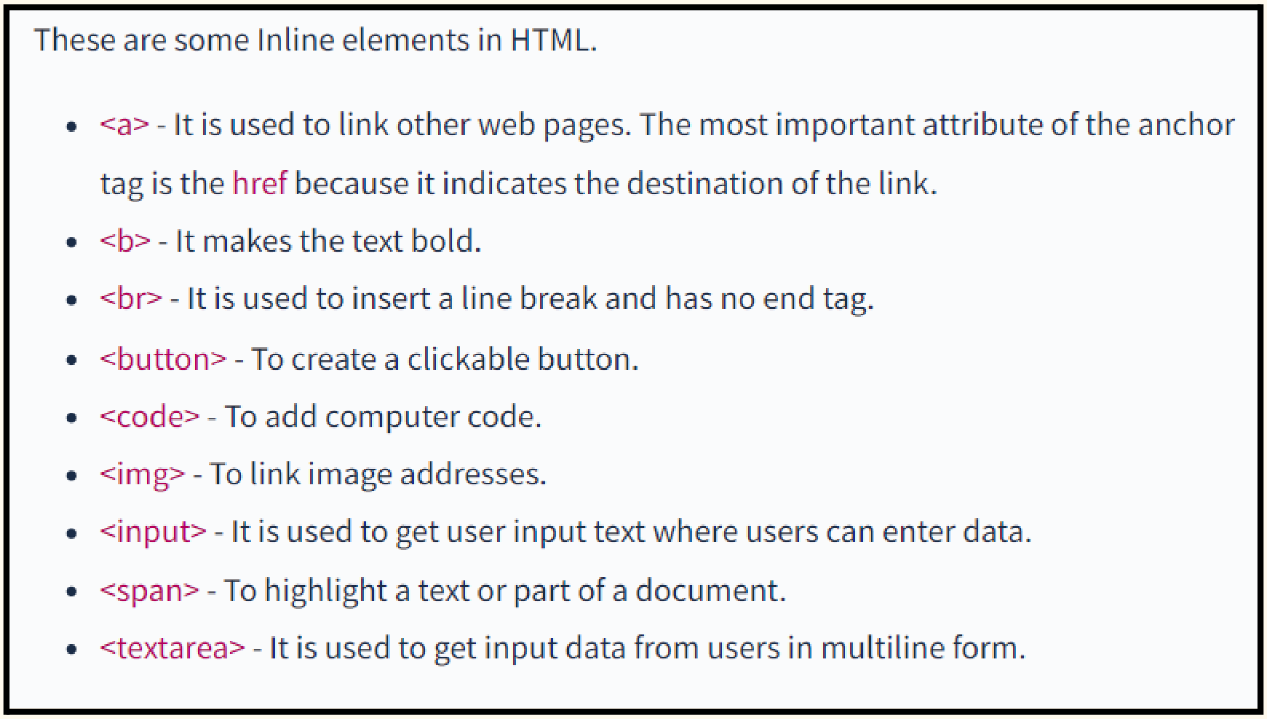


These are some supported tags of Block elements:

- <article> - Self- contained and independent content.
- <aside> - The content inside aside is often placed at the sidebar in a document.
- <div> - Container for HTML elements.
- <fieldset> - Group the same or related items.
- <figcaption> - Define the caption for <figure> element.
- <figure> - Contain content like illustrations, figures, images etc.
- <footer> - It defines the footer of the section.
- <form> - Get information from the user input.
- <h1>-<h6> - Define HTML headings, where h1 is largest and h6 is smallest.
- <header> - Container of introduction.
- <hr> - Separate content using horizontal lines.
- <li> - Add list items, ordered(<ol>) or unordered lists (<ul>).
- <main> - Add the main content we used the main tag.
- <nav> - Add navigation links.
- <section>- Add a section.
- <table> - Add a table.

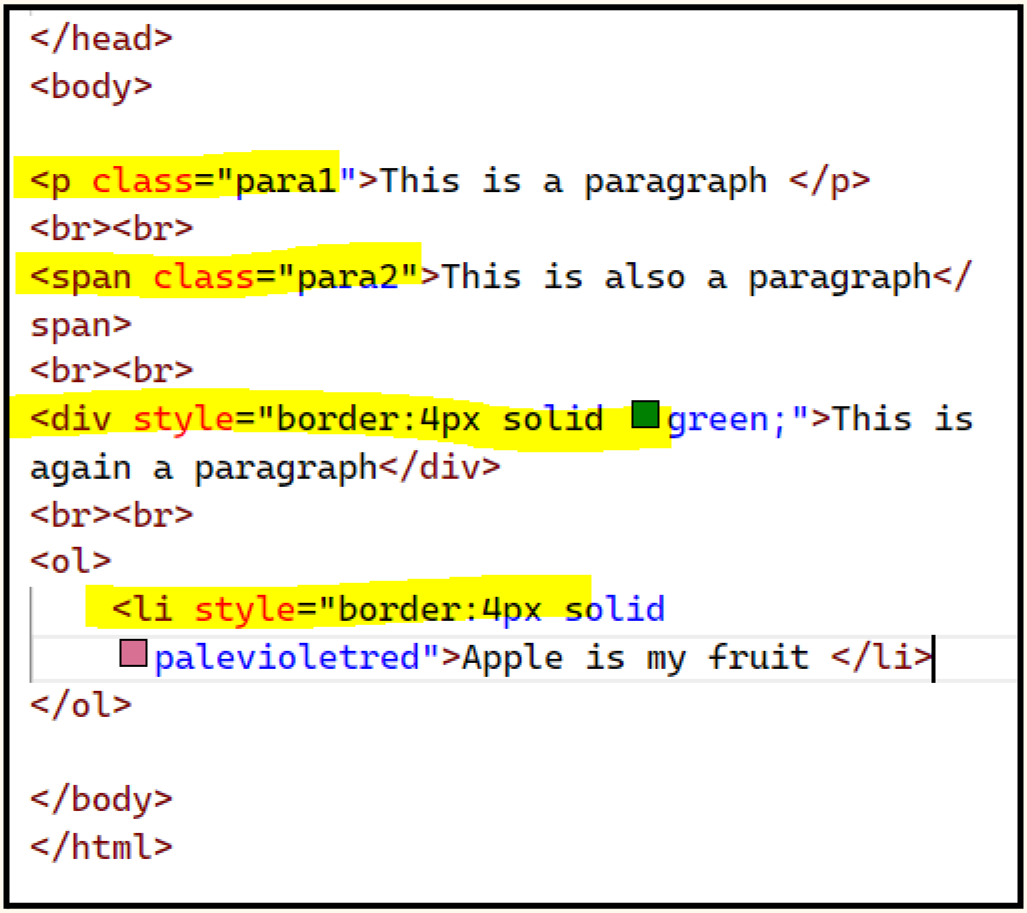
Inline Elements

Inline elements never start from a new line and only cover the width according to the size of bounded tags in the HTML element.

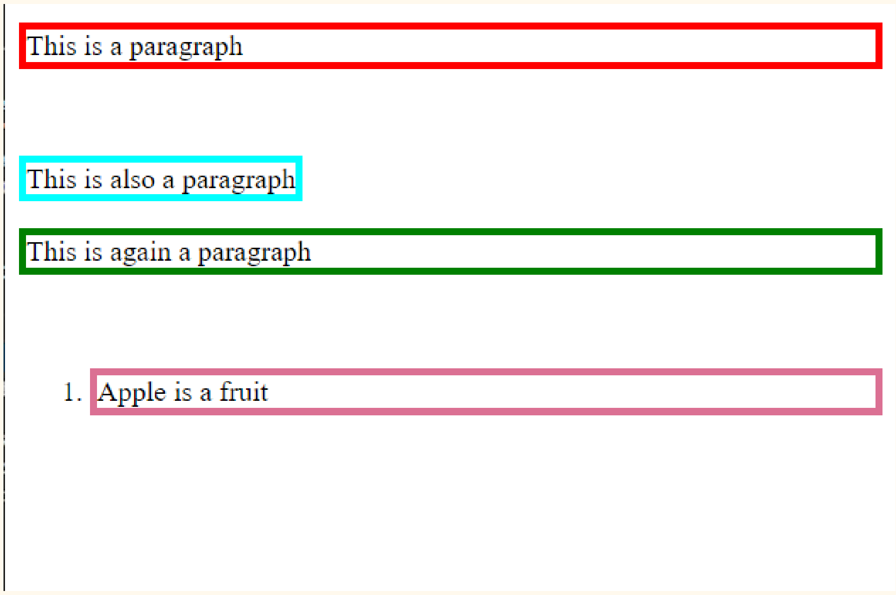


These are some Inline elements in HTML.

- <a> - It is used to link other web pages. The most important attribute of the anchor tag is the href because it indicates the destination of the link.
- <b> - It makes the text bold.
-
 - It is used to insert a line break and has no end tag.
- <button> - To create a clickable button.
- <code> - To add computer code.
- <img> - To link image addresses.
- <input> - It is used to get user input text where users can enter data.
- <span> - To highlight a text or part of a document.
- <textarea> - It is used to get input data from users in multiline form.



```
</head>
<body>

<p class="para1">This is a paragraph </p>
<br><br>
<span class="para2">This is also a paragraph</
span>
<br><br>
<div style="border:4px solid 🟩green;">This is
again a paragraph</div>
<br><br>
<ol>
    <li style="border:4px solid
     🟥palevioletred">Apple is my fruit </li>
</ol>

</body>
</html>
```

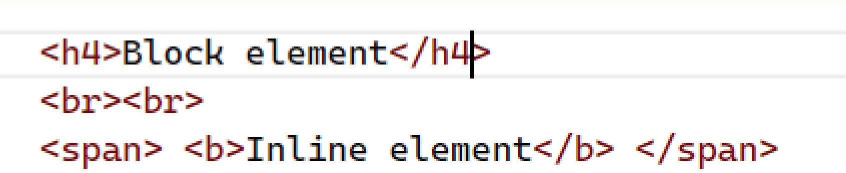


This is a paragraph

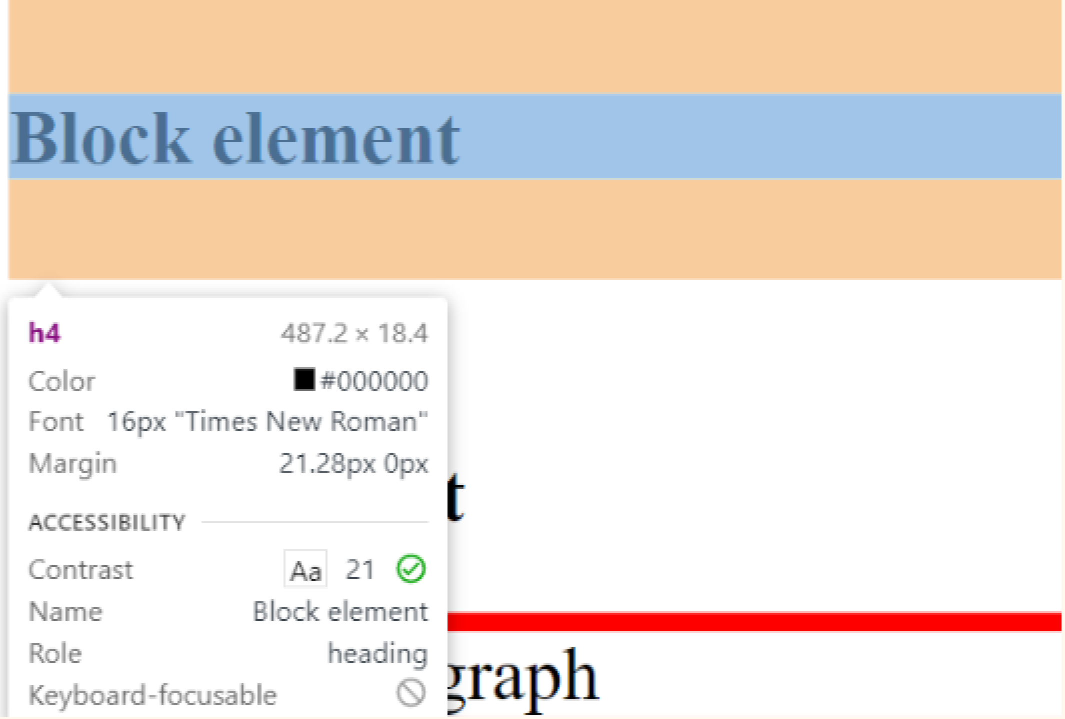
This is also a paragraph

This is again a paragraph

1. Apple is a fruit



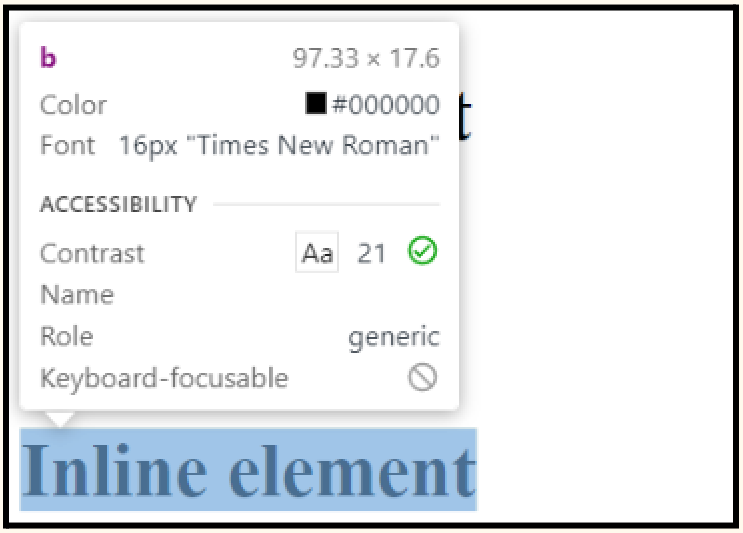
```
<h4>Block element</h4>
<br><br>
<span> <b>Inline element</b> </span>
```



Block element

h4 487.2 × 18.4
Color ■#000000
Font 16px "Times New Roman"
Margin 21.28px 0px

ACCESSIBILITY
Contrast Aa 21 ✓
Name Block element
Role heading
Keyboard-focusable ⃠



b 97.33 × 17.6
Color ■#000000
Font 16px "Times New Roman"

ACCESSIBILITY
Contrast Aa 21 ✓
Name
Role generic
Keyboard-focusable ⃠

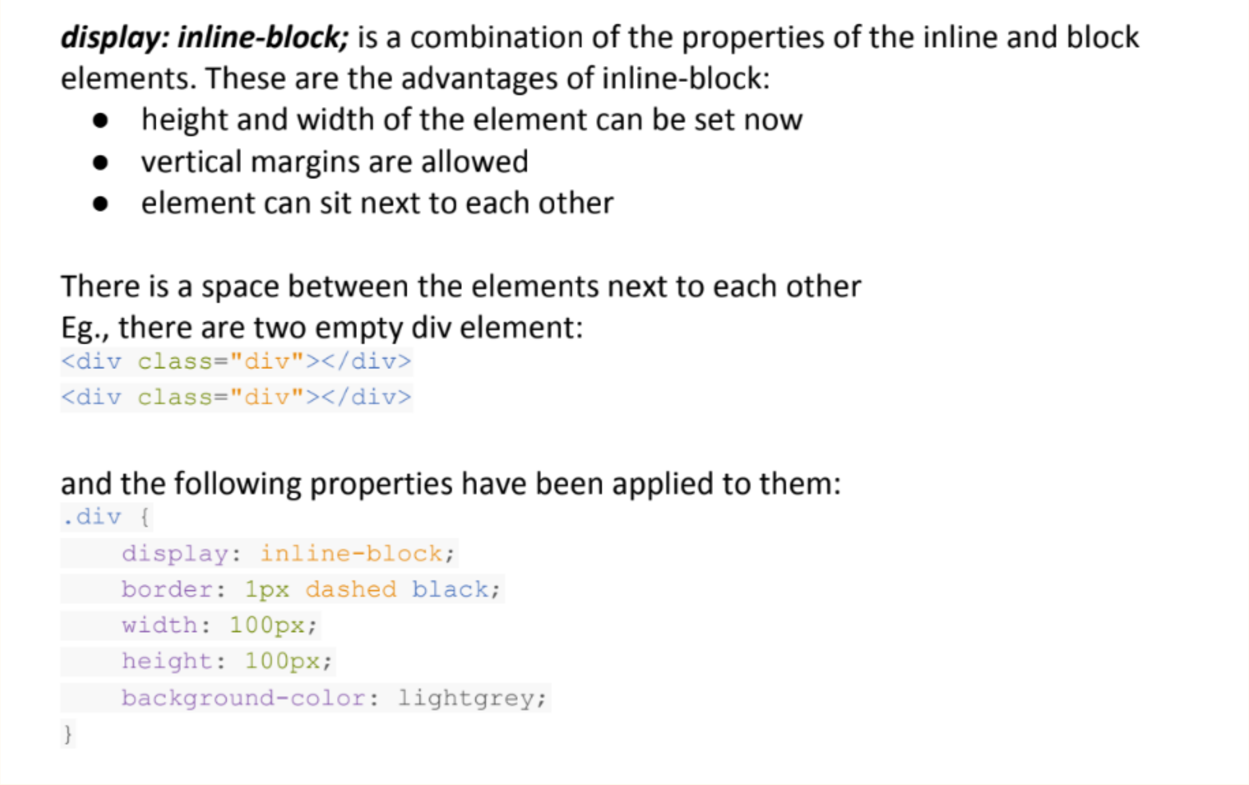
Inline element

Inline-Block:

Compared to display: inline, the major difference is that display: inline-block allows setting a width and height on the element.

Also, with display: inline-block, the top and bottom margins/paddings are respected, but with display: inline they are not.

Compared to display: block, the major difference is that display: inline-block does not add a line-break after the element, so the element can sit next to other elements.



display: inline-block; is a combination of the properties of the inline and block elements. These are the advantages of inline-block:

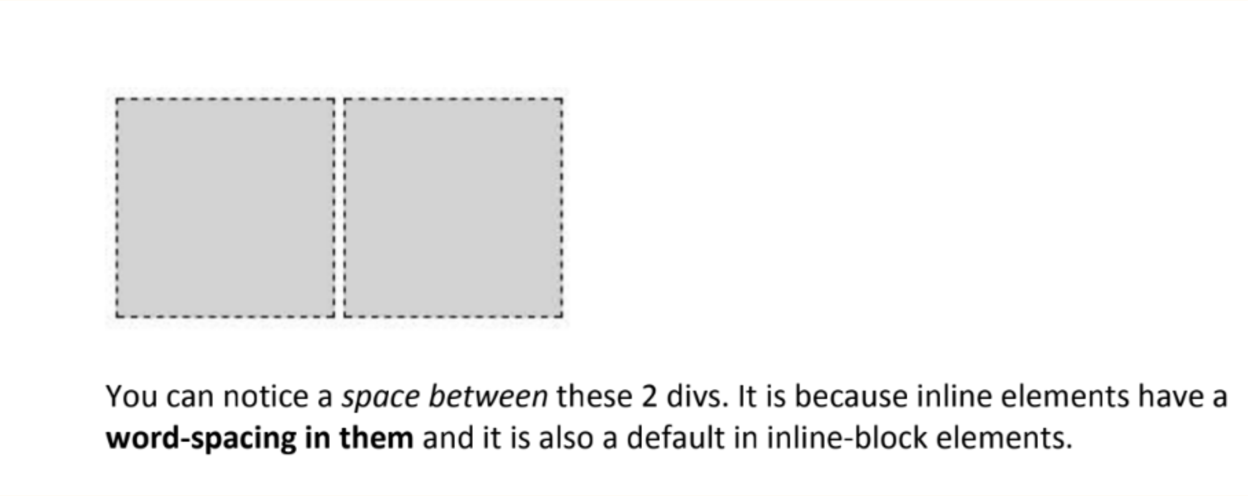
- height and width of the element can be set now
- vertical margins are allowed
- element can sit next to each other

There is a space between the elements next to each other
Eg., there are two empty div element:

```
<div class="div"></div>
<div class="div"></div>
```

and the following properties have been applied to them:

```
.div {
    display: inline-block;
    border: 1px dashed black;
    width: 100px;
    height: 100px;
    background-color: lightgrey;
}
```



You can notice a *space between* these 2 divs. It is because inline elements have a **word-spacing in them** and it is also a default in inline-block elements.

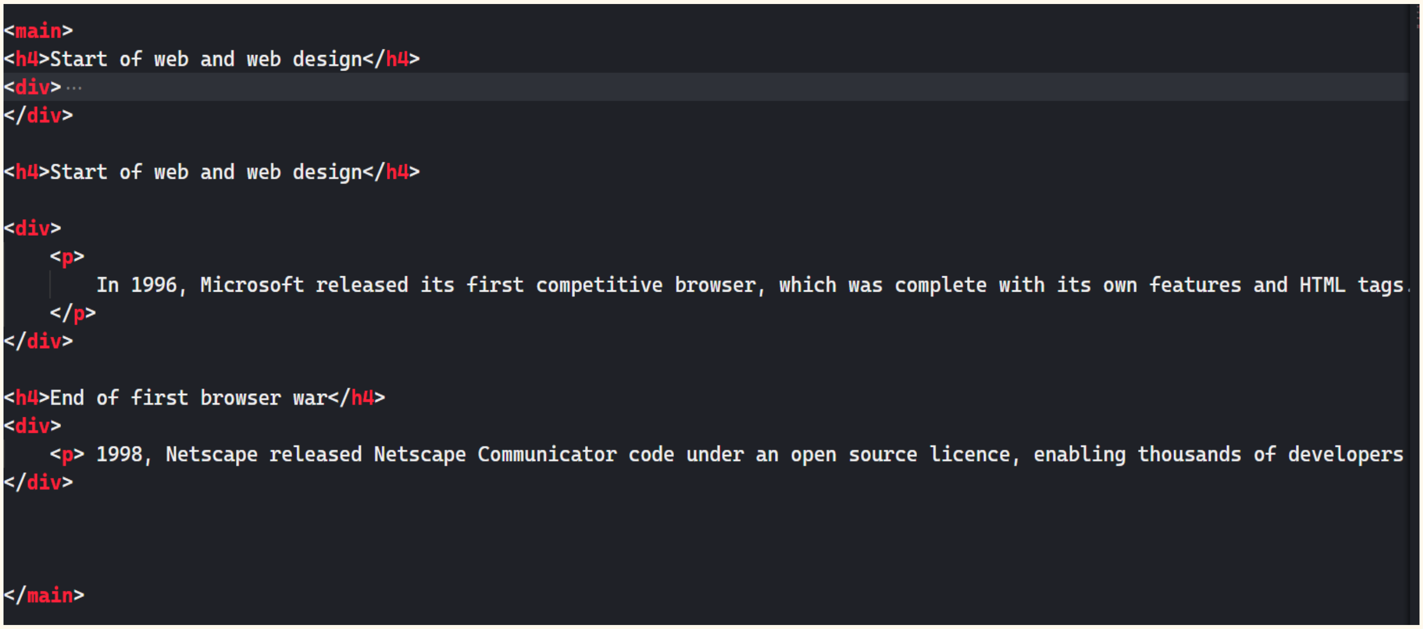
->> Aside Section/ Aside tag <<-

The <aside> tag defines some content aside from the content it is placed in.

The aside content should be indirectly related to the surrounding content.

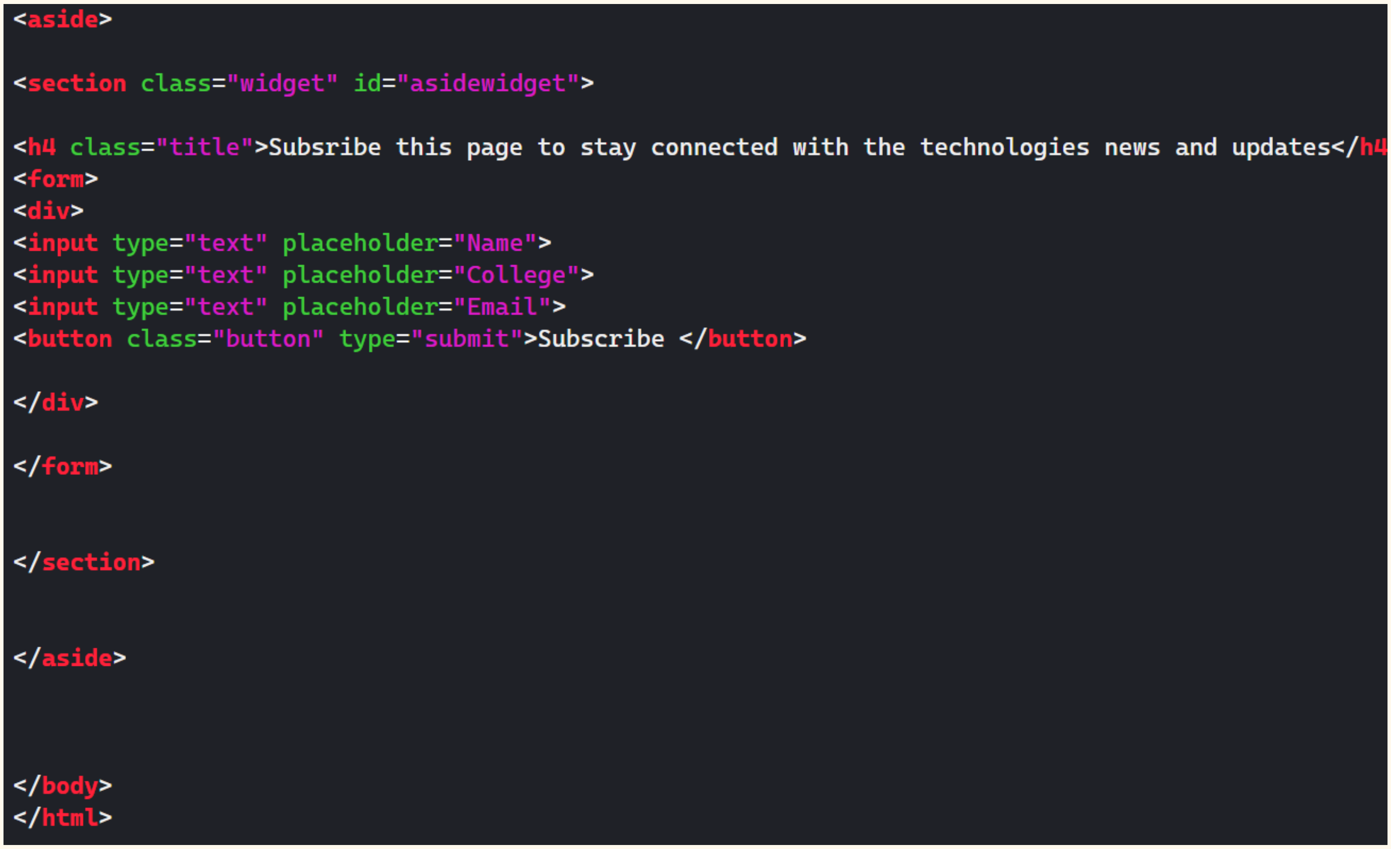
Tip: The <aside> content is often placed as a sidebar in a document.

MAIN SECTION:



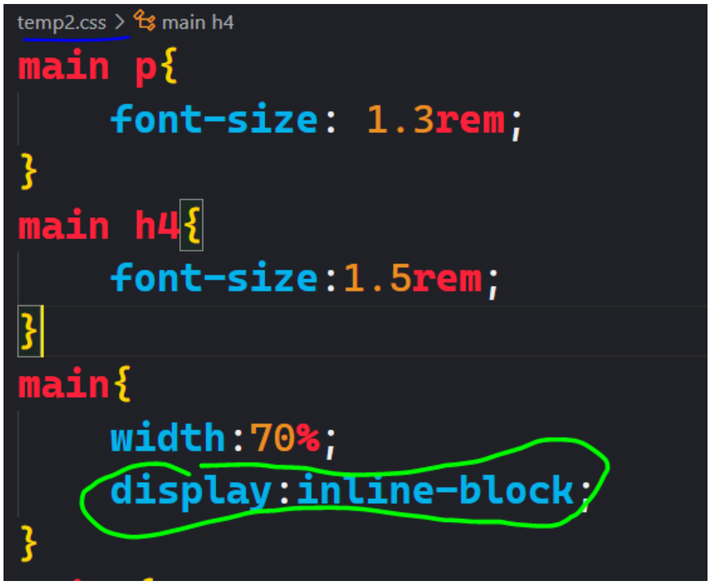
```html
<main>
<h4>Start of web and web design</h4>
<div>···
</div>

<h4>Start of web and web design</h4>

<div>
    <p>
        In 1996, Microsoft released its first competitive browser, which was complete with its own features and HTML tags.
    </p>
</div>

<h4>End of first browser war</h4>
<div>
    <p> 1998, Netscape released Netscape Communicator code under an open source licence, enabling thousands of developers
</div>



</main>
```

ASIDE SECTION:

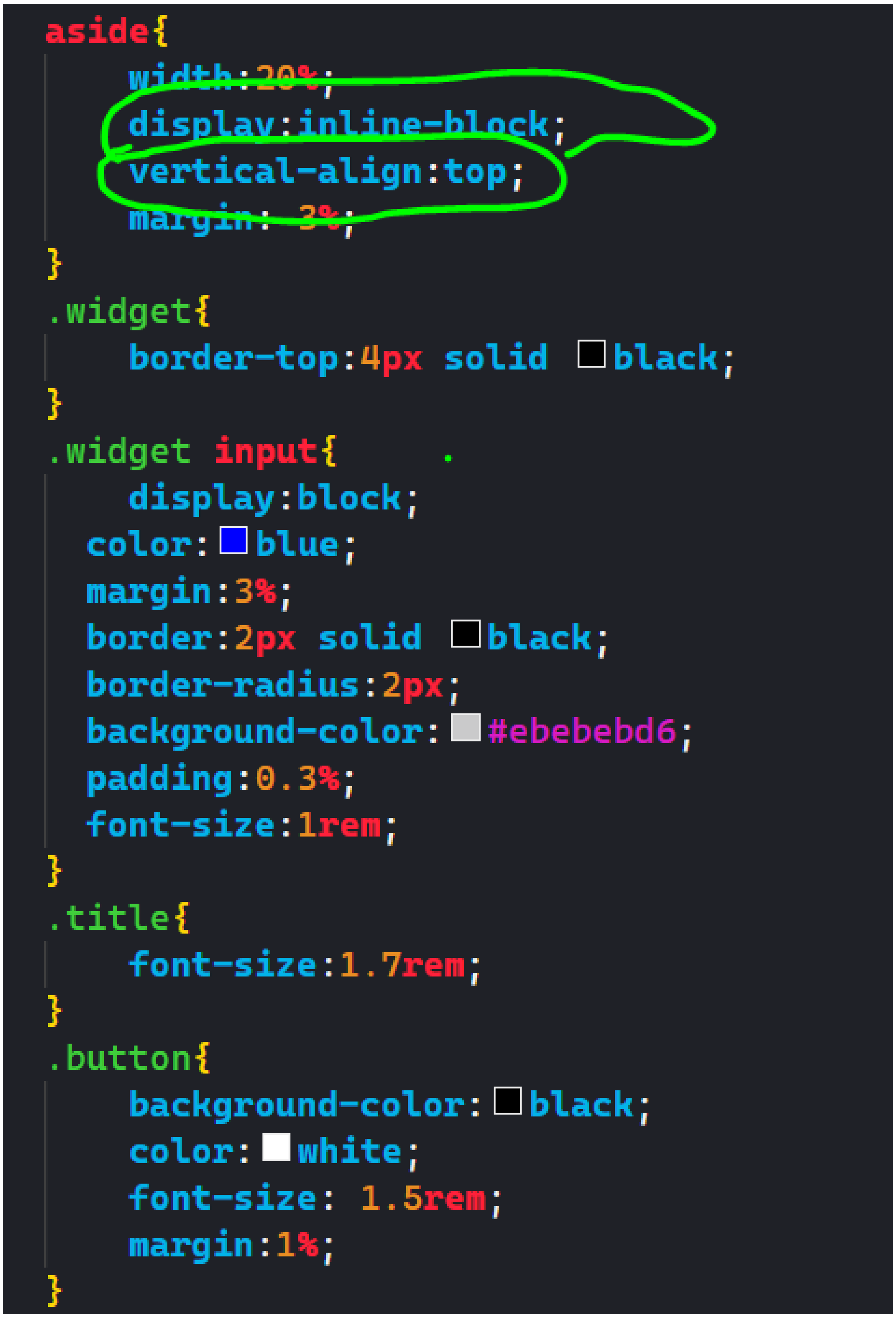


```html
<aside>

<section class="widget" id="asidewidget">

<h4 class="title">Subsribe this page to stay connected with the technologies news and updates</h4
<form>
<div>
<input type="text" placeholder="Name">
<input type="text" placeholder="College">
<input type="text" placeholder="Email">
<button class="button" type="submit">Subscribe </button>

</div>

</form>


</section>


</aside>




</body>
</html>
```

CSS FILE:

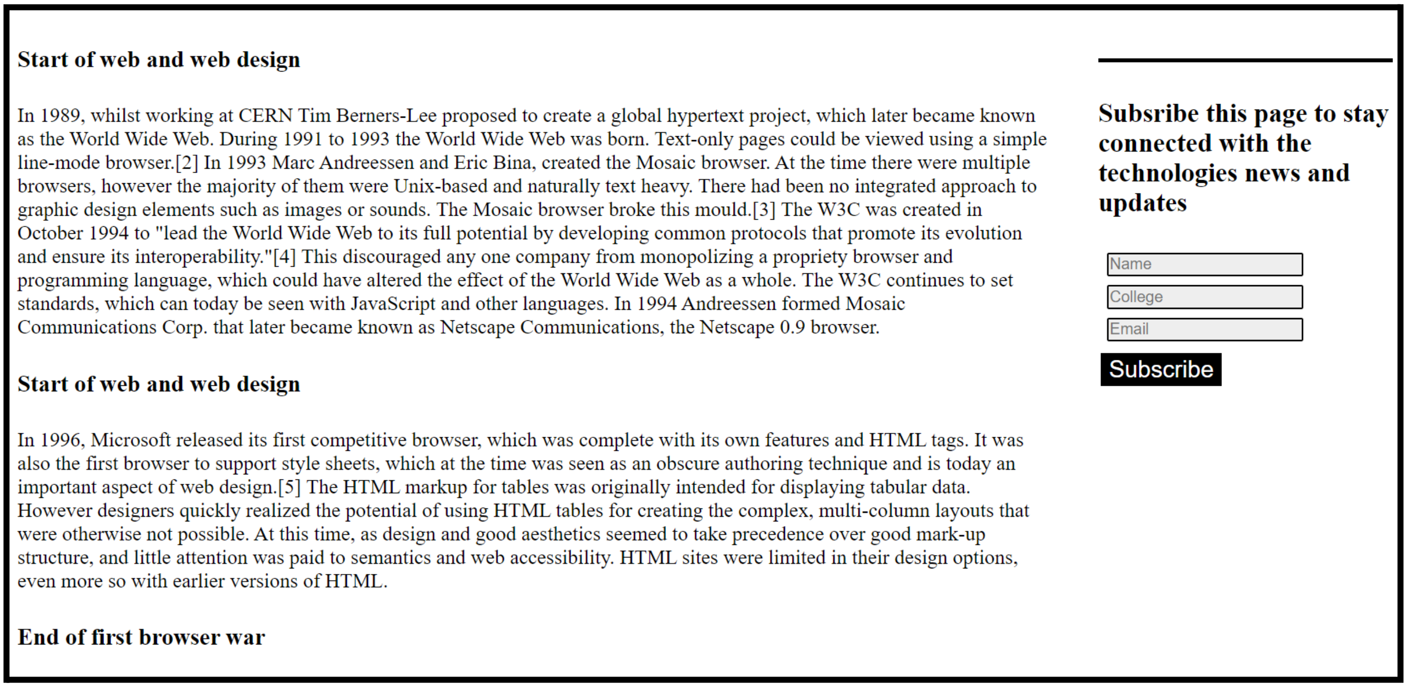


```
main p{
    font-size: 1.3rem;
}
main h4{
    font-size:1.5rem;
}
main{
    width:70%;
    display:inline-block;
}
```



```
aside{
    width:20%;
    display:inline-block;
    vertical-align:top;
    margin: 3%;
}
.widget{
    border-top:4px solid ■black;
}
.widget input{
    display:block;
  color:■blue;
  margin:3%;
  border:2px solid ■black;
  border-radius:2px;
  background-color:■#ebebebd6;
  padding:0.3%;
  font-size:1rem;
}
.title{
    font-size:1.7rem;
}
.button{
    background-color:■black;
    color:□white;
    font-size: 1.5rem;
    margin:1%;
}
```

OUTPUT SCREEN:-



Start of web and web design

In 1989, whilst working at CERN Tim Berners-Lee proposed to create a global hypertext project, which later became known as the World Wide Web. During 1991 to 1993 the World Wide Web was born. Text-only pages could be viewed using a simple line-mode browser.[2] In 1993 Marc Andreessen and Eric Bina, created the Mosaic browser. At the time there were multiple browsers, however the majority of them were Unix-based and naturally text heavy. There had been no integrated approach to graphic design elements such as images or sounds. The Mosaic browser broke this mould.[3] The W3C was created in October 1994 to "lead the World Wide Web to its full potential by developing common protocols that promote its evolution and ensure its interoperability."[4] This discouraged any one company from monopolizing a propriety browser and programming language, which could have altered the effect of the World Wide Web as a whole. The W3C continues to set standards, which can today be seen with JavaScript and other languages. In 1994 Andreessen formed Mosaic Communications Corp. that later became known as Netscape Communications, the Netscape 0.9 browser.

Start of web and web design

In 1996, Microsoft released its first competitive browser, which was complete with its own features and HTML tags. It was also the first browser to support style sheets, which at the time was seen as an obscure authoring technique and is today an important aspect of web design.[5] The HTML markup for tables was originally intended for displaying tabular data. However designers quickly realized the potential of using HTML tables for creating the complex, multi-column layouts that were otherwise not possible. At this time, as design and good aesthetics seemed to take precedence over good mark-up structure, and little attention was paid to semantics and web accessibility. HTML sites were limited in their design options, even more so with earlier versions of HTML.

End of first browser war

Subsribe this page to stay connected with the technologies news and updates

->>>> Position property <<<-

The CSS position property defines the position of an HTML element inside the viewport. The position property is used with left, right, top, bottom, and z-index properties to position the elements in the viewport.

Values of CSS position property

The CSS position property accepts five values used to determine the position of an HTML element.

->static

->relative

->absolute

->fixed

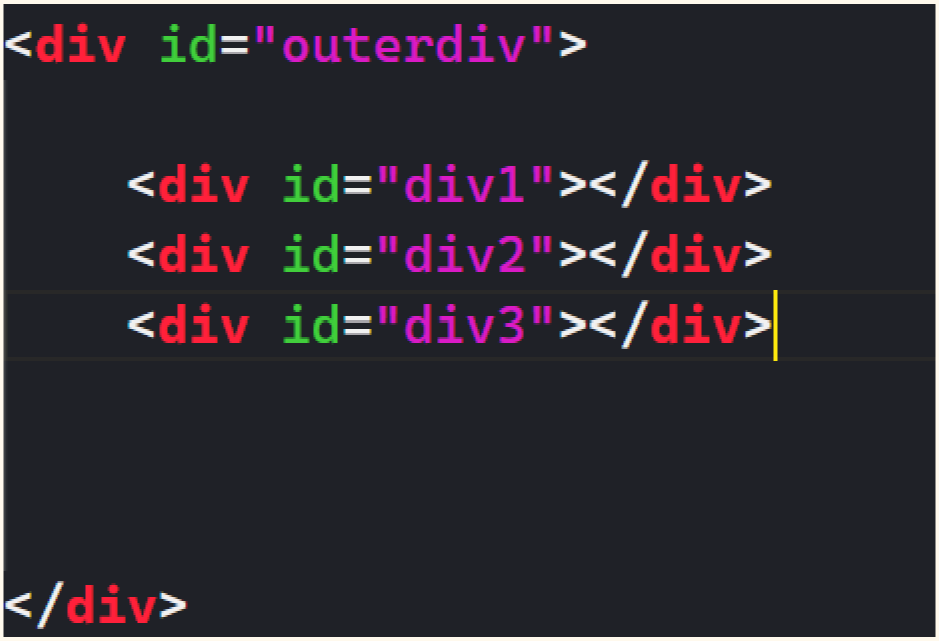
->sticky

1.Static:

All HTML elements are positioned static by default. With static positioning, the elements are positioned along with the natural flow of the document. The properties left, right, top, bottom, and z-index do not affect the elements positioned as static

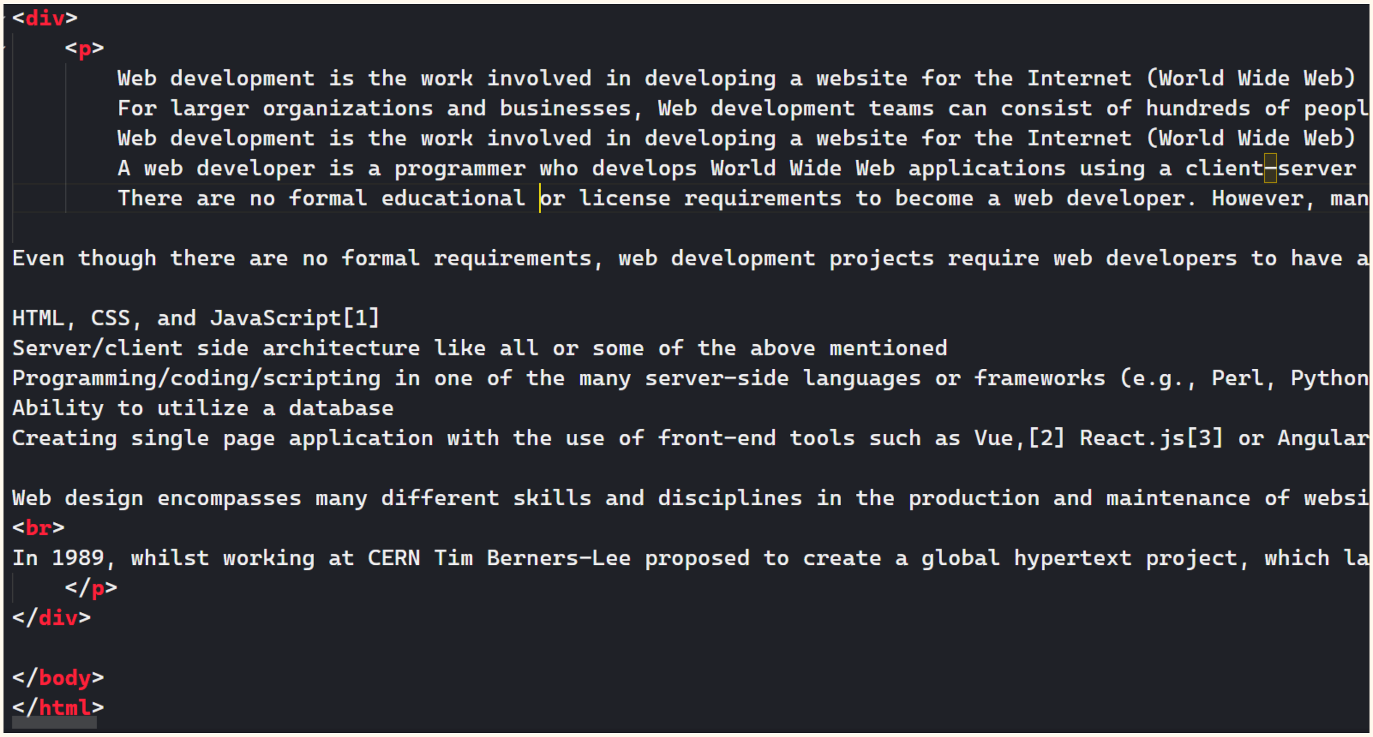
2.Relative

Let’s create 3 div:



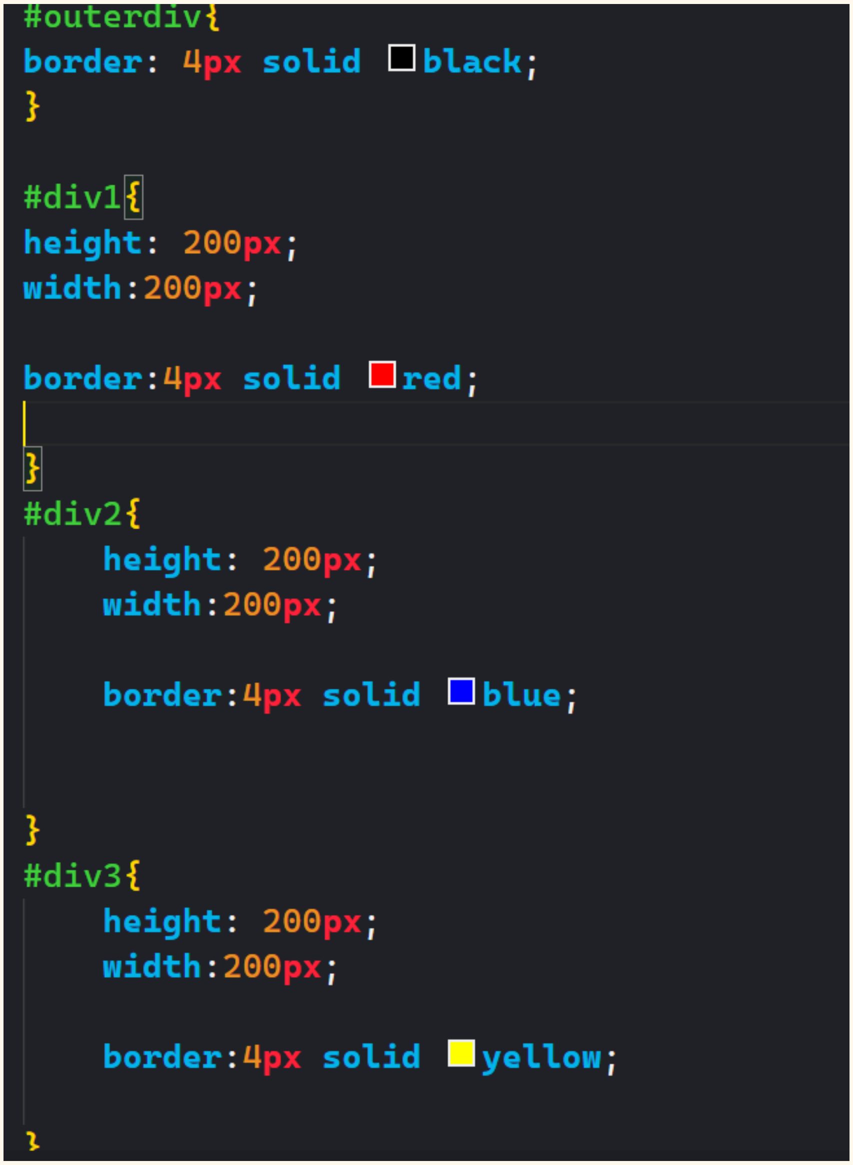
```
<div id="outerdiv">

    <div id="div1"></div>
    <div id="div2"></div>
    <div id="div3"></div>



</div>
```

And 1 div for some content

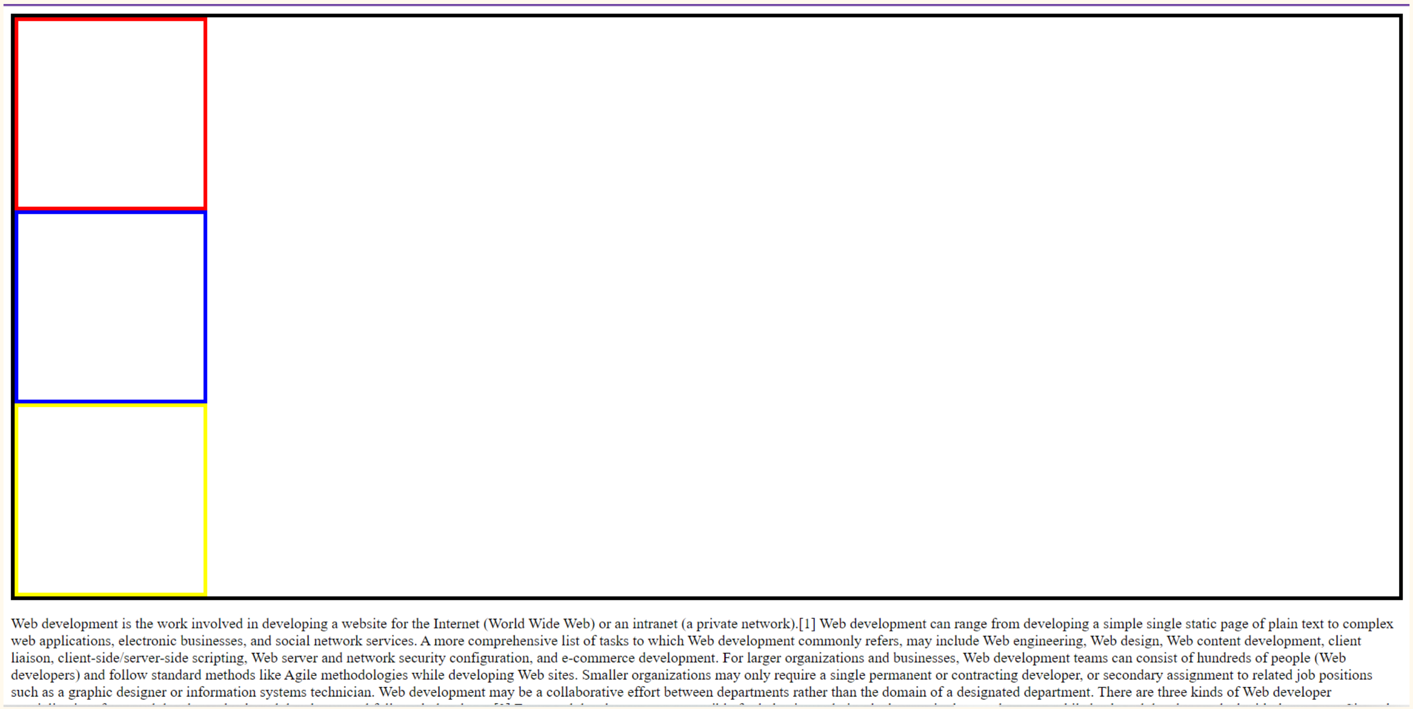


```
<div>
    <p>
        Web development is the work involved in developing a website for the Internet (World Wide Web)
        For larger organizations and businesses, Web development teams can consist of hundreds of peopl
        Web development is the work involved in developing a website for the Internet (World Wide Web)
        A web developer is a programmer who develops World Wide Web applications using a client-server
        There are no formal educational or license requirements to become a web developer. However, man

Even though there are no formal requirements, web development projects require web developers to have a

HTML, CSS, and JavaScript[1]
Server/client side architecture like all or some of the above mentioned
Programming/coding/scripting in one of the many server-side languages or frameworks (e.g., Perl, Python
Ability to utilize a database
Creating single page application with the use of front-end tools such as Vue,[2] React.js[3] or Angular

Web design encompasses many different skills and disciplines in the production and maintenance of websi
<br>
In 1989, whilst working at CERN Tim Berners-Lee proposed to create a global hypertext project, which la
    </p>
</div>

</body>
</html>
```

And in CSS part:-



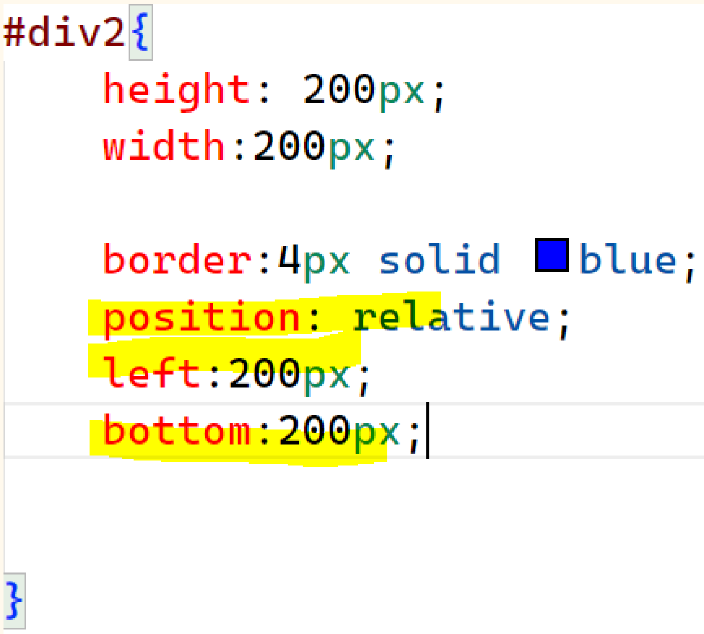
```css
#outerdiv{
border: 4px solid black;
}

#div1{
height: 200px;
width:200px;

border:4px solid red;

}
#div2{
    height: 200px;
    width:200px;

    border:4px solid blue;


}
#div3{
    height: 200px;
    width:200px;

    border:4px solid yellow;

}
```



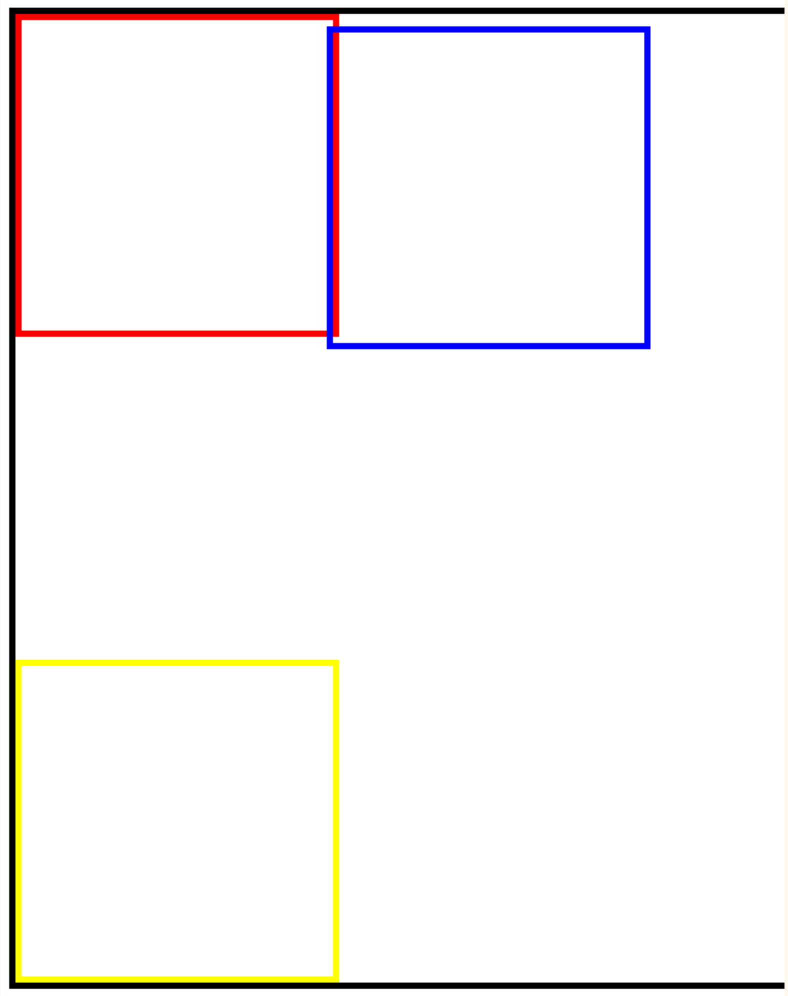
Web development is the work involved in developing a website for the Internet (World Wide Web) or an intranet (a private network).[1] Web development can range from developing a simple single static page of plain text to complex web applications, electronic businesses, and social network services. A more comprehensive list of tasks to which Web development commonly refers, may include Web engineering, Web design, Web content development, client liaison, client-side/server-side scripting, Web server and network security configuration, and e-commerce development. For larger organizations and businesses, Web development teams can consist of hundreds of people (Web developers) and follow standard methods like Agile methodologies while developing Web sites. Smaller organizations may only require a single permanent or contracting developer, or secondary assignment to related job positions such as a graphic designer or information systems technician. Web development may be a collaborative effort between departments rather than the domain of a designated department. There are three kinds of Web developer

With relative positioning, the elements are positioned along with the natural flow of the document. But unlike static elements, the position of the relative elements can be modified using left, right, top, bottom, and z-index properties.

# In this the element move relative to its original position / default position



```
#div2{
    height: 200px;
    width:200px;

    border:4px solid  blue;
    position: relative;
    left:200px;
    bottom:200px;


}
```

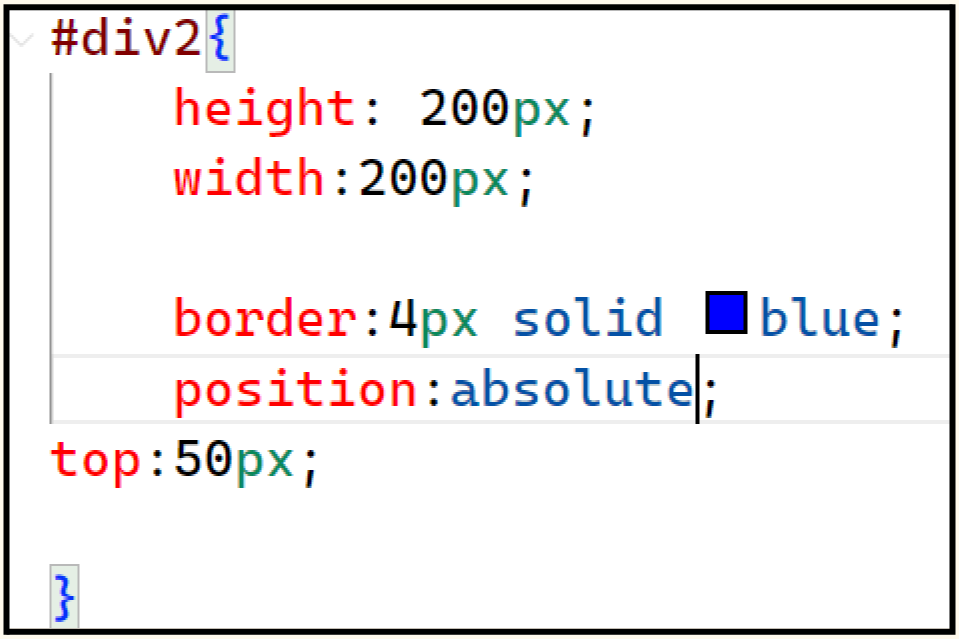


*****Note*****

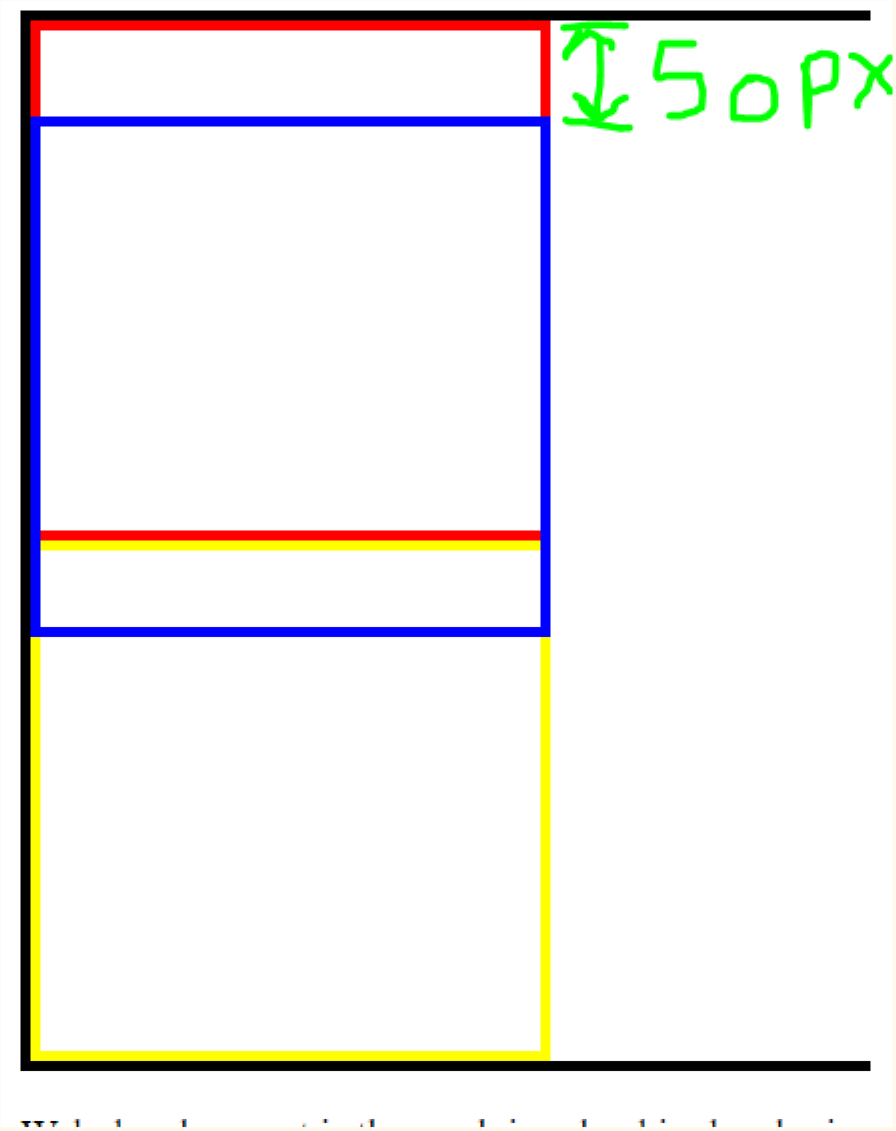
The space of div 2 remains unoccupied

3. Absolute position

The elements are positioned relative to their parent elements with absolute positioning. The absolute elements are positioned relative to the closest ancestor (or closest parent element) with any position property other than static. If the closest ancestor has a static position, the element is positioned relative to the next parent element without the static position property.



```
#div2{
    height: 200px;
    width:200px;

    border:4px solid blue;
    position:absolute;
top:50px;

}
```



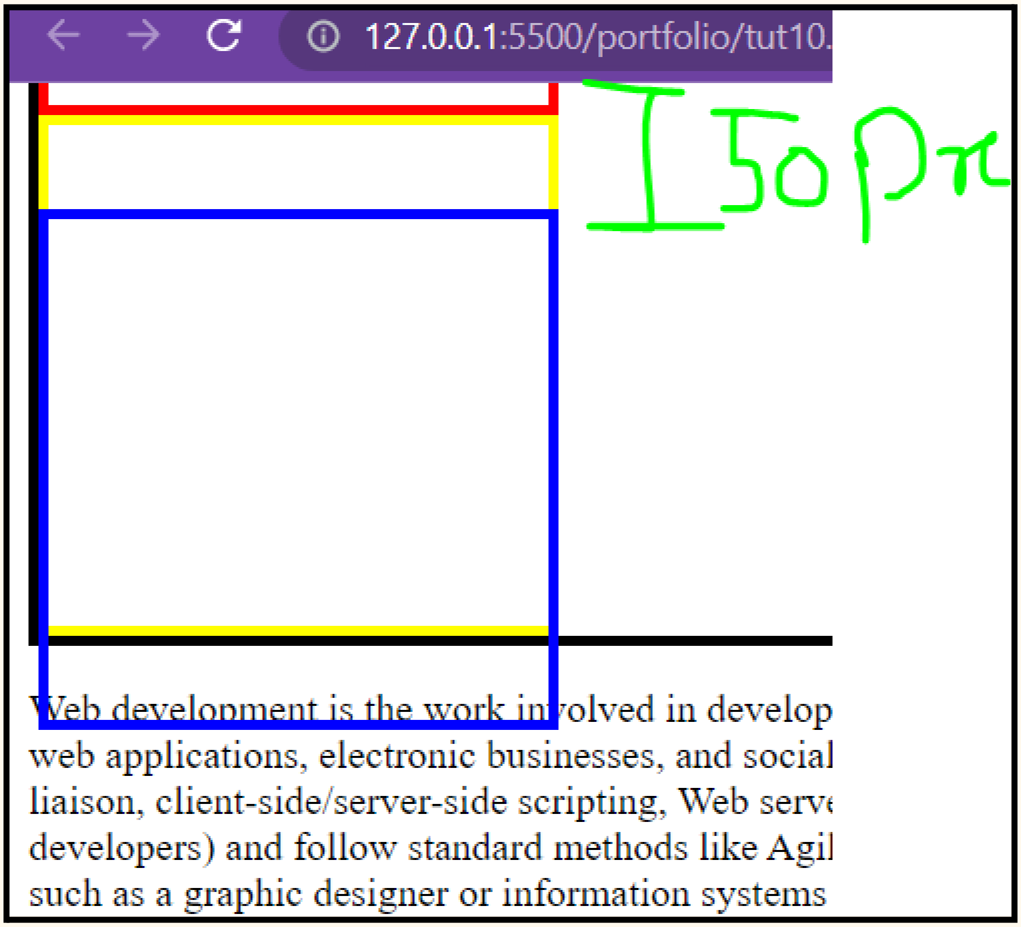
The div shifts 50 px from the top of the border as the nearest ancestor of div2 is div with class=”outerdiv”

*****Note*****

The space of div 2 occupied by the other element

4. Fixed position:-

Elements with fixed positioning are always positioned relative to the HTML element (i.e.) the root of the document. The fixed elements stay in the same position irrespective of the scrolling. Like absolute elements, the fixed elements are also removed from the natural flow of the document, and other elements occupy their place.

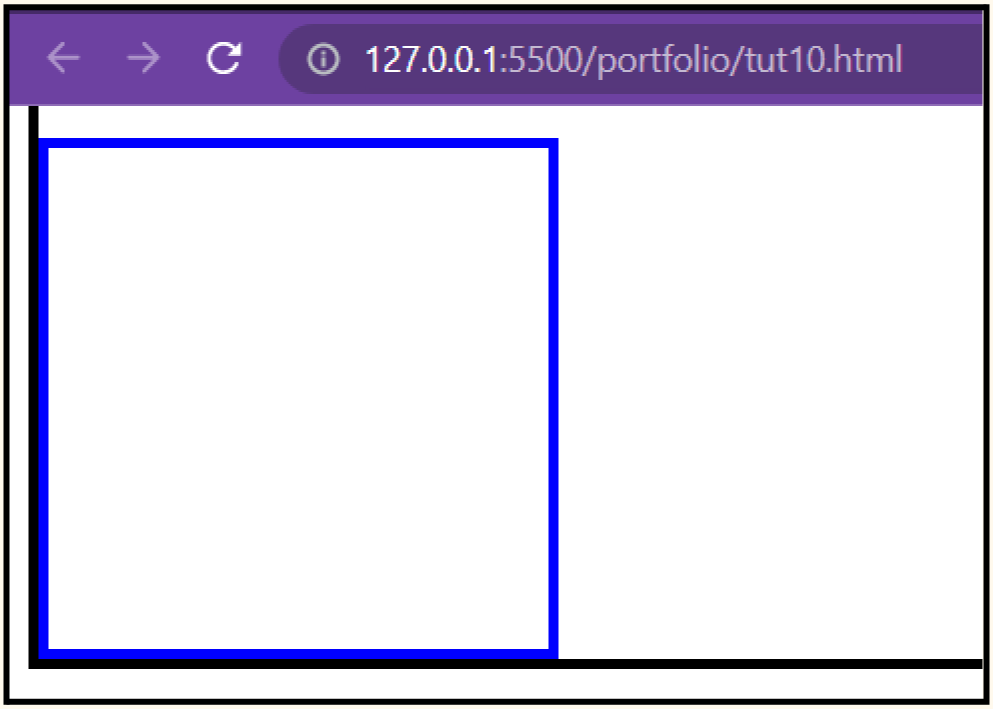


Web development is the work involved in develop
web applications, electronic businesses, and social
liaison, client-side/server-side scripting, Web serve
developers) and follow standard methods like Agil
such as a graphic designer or information systems

The div shifts 50 px from the top of the view-port and the div 2 element remains freezed also the position was occupied by other elements.

5.Sticky position:-

With sticky positioning, the element behaves like a relative positioned until a certain scroll point, and then it will be fixed.

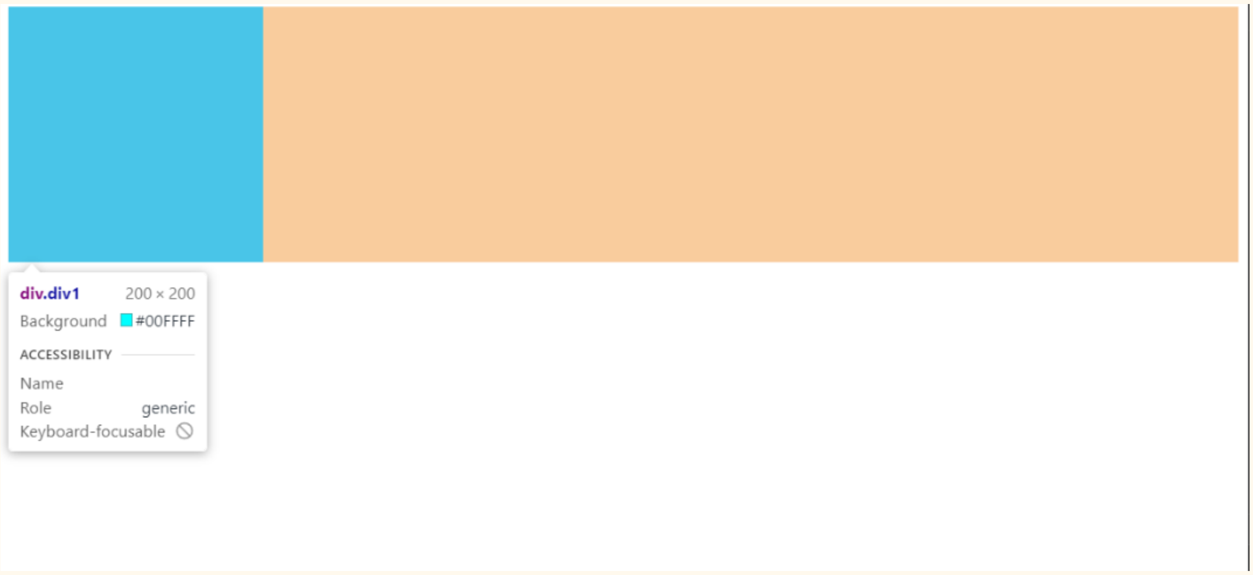


https://scaler.com/topics/images/position-property-in-css-image7.gif

->>>> Box Model <<<<-;

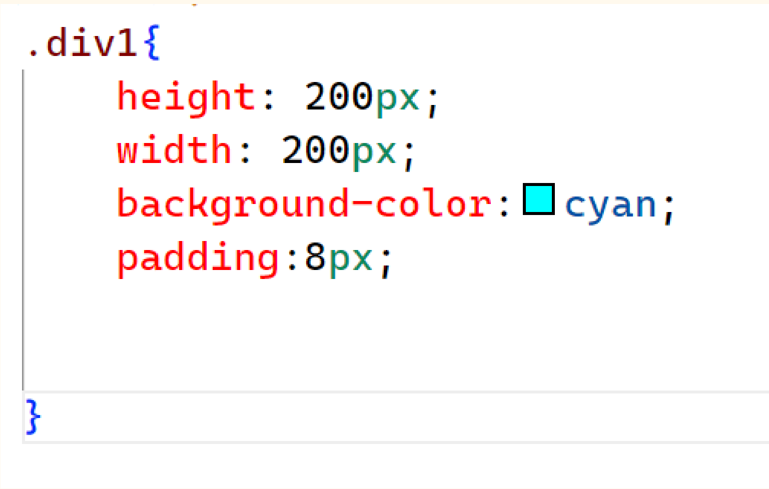
There are two types of model namely box model and content model

1.Content Model

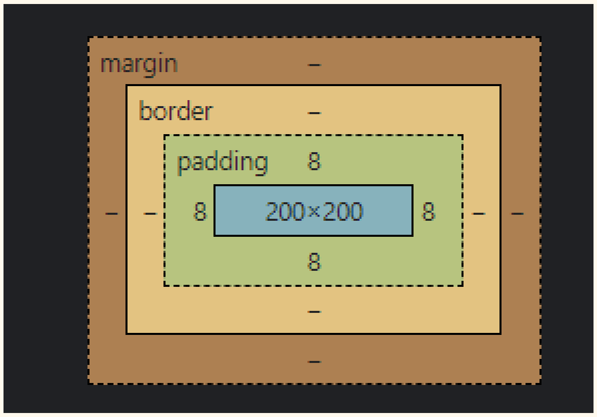


div.div1 200 × 200
Background ■#00FFFF
ACCESSIBILITY
Name
Role generic
Keyboard-focusable ⃠

Here the height and width are of the content box it means exclusive of padding and border.

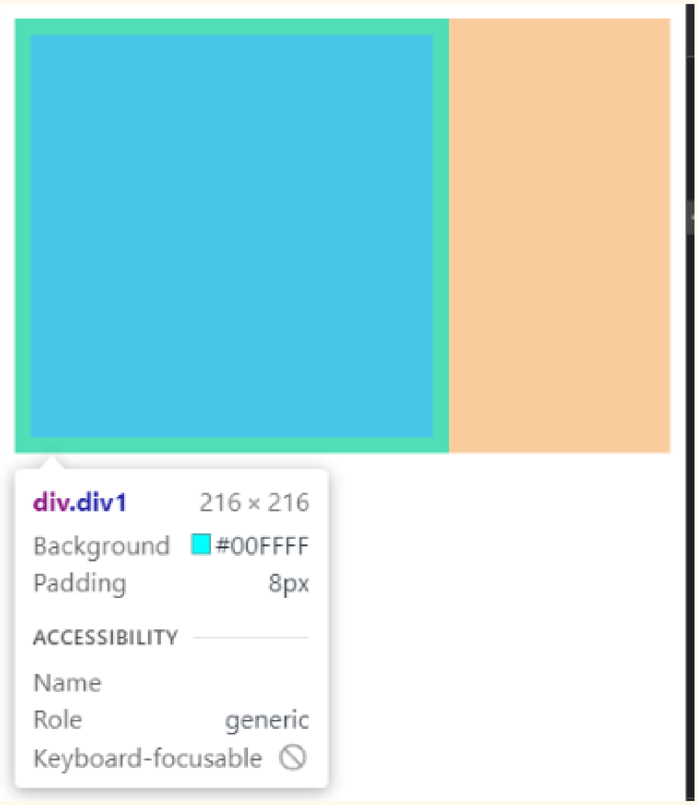


```
.div1{
    height: 200px;
    width: 200px;
    background-color: cyan;
    padding:8px;


}
```



Size of the content box is still 200px but size of div become 216*216

216 = content(200) + padding(16)



div.div1 216 × 216

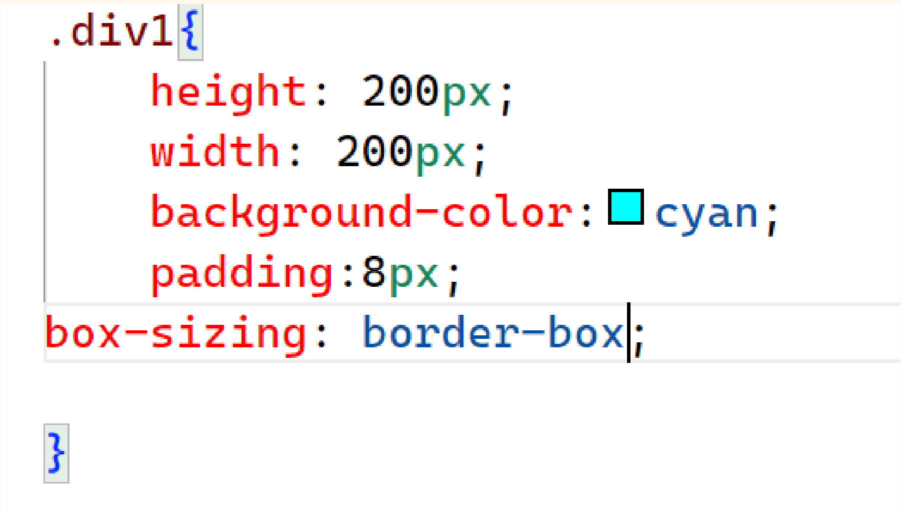
Background	#00FFFF
Padding	8px

ACCESSIBILITY

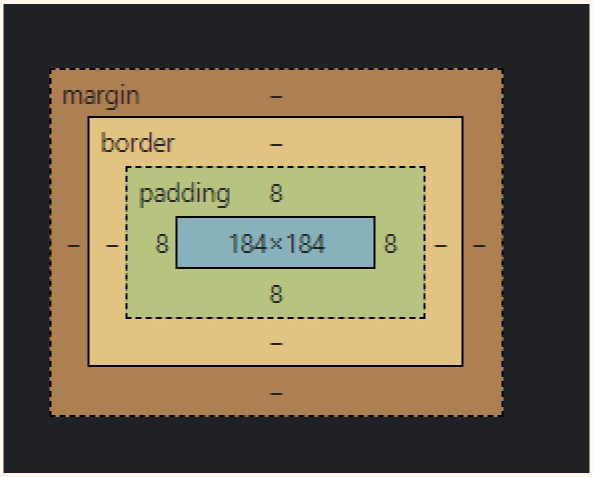
Name	
Role	generic
Keyboard-focusable	⃠

“By default the box model is Content model”

2.Box Model



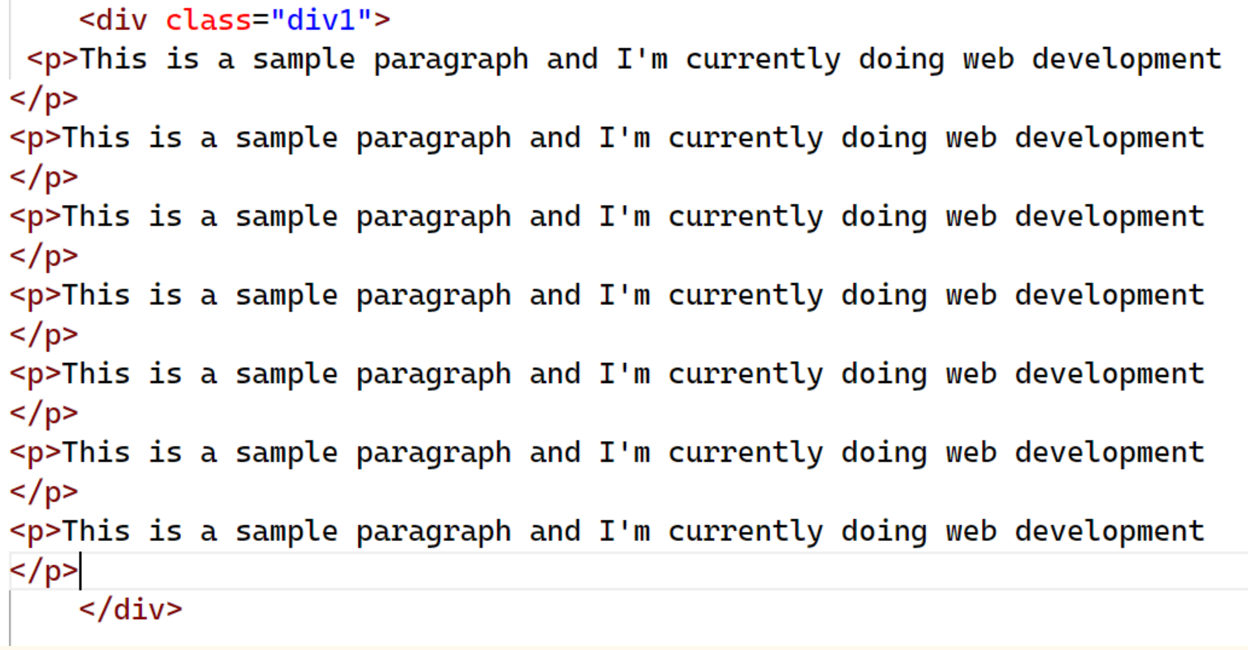
```
.div1{
    height: 200px;
    width: 200px;
    background-color: cyan;
    padding:8px;
box-sizing: border-box;

}
```



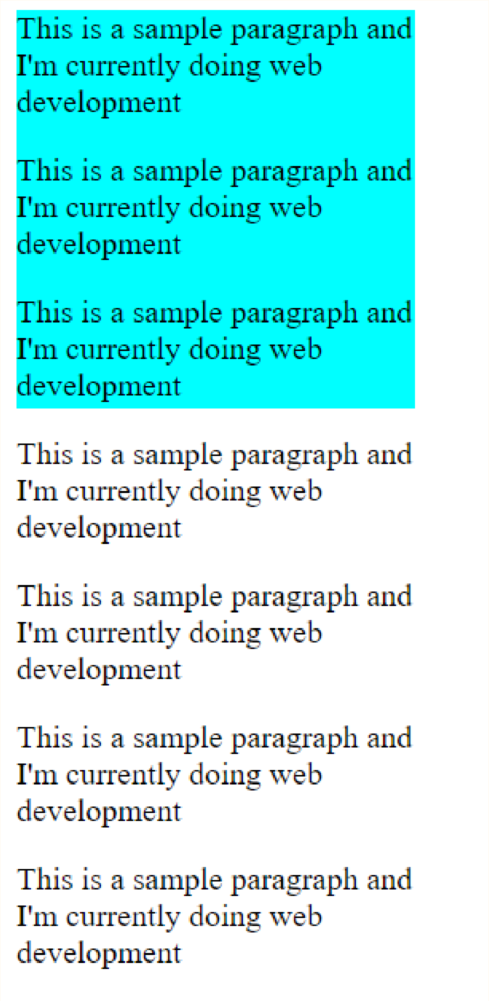
Here the size 200*200 is not for the content box but for the entire div.

200px= content(184) + padding(16)

->>>>Overflow<<<-;



```
    <div class="div1">
 <p>This is a sample paragraph and I'm currently doing web development
</p>
<p>This is a sample paragraph and I'm currently doing web development
</p>
<p>This is a sample paragraph and I'm currently doing web development
</p>
<p>This is a sample paragraph and I'm currently doing web development
</p>
<p>This is a sample paragraph and I'm currently doing web development
</p>
<p>This is a sample paragraph and I'm currently doing web development
</p>
<p>This is a sample paragraph and I'm currently doing web development
</p>
    </div>
```



This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

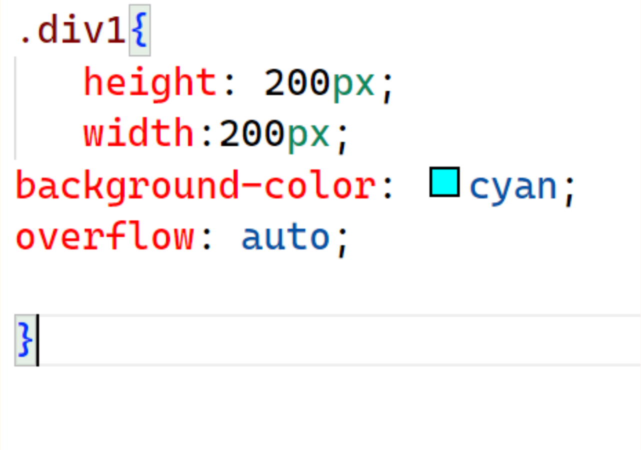
This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

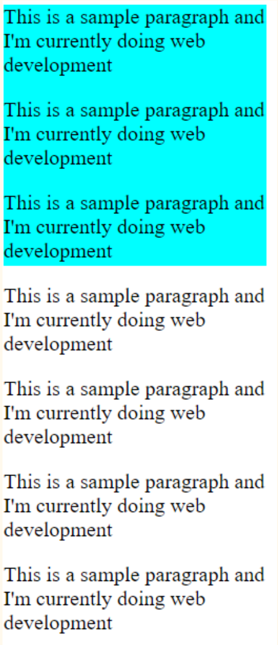
This is a sample paragraph and I'm currently doing web development

Different properties of Overflow:

1. auto



```css
.div1{
    height: 200px;
    width:200px;
background-color: cyan;
overflow: auto;

}
```



This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

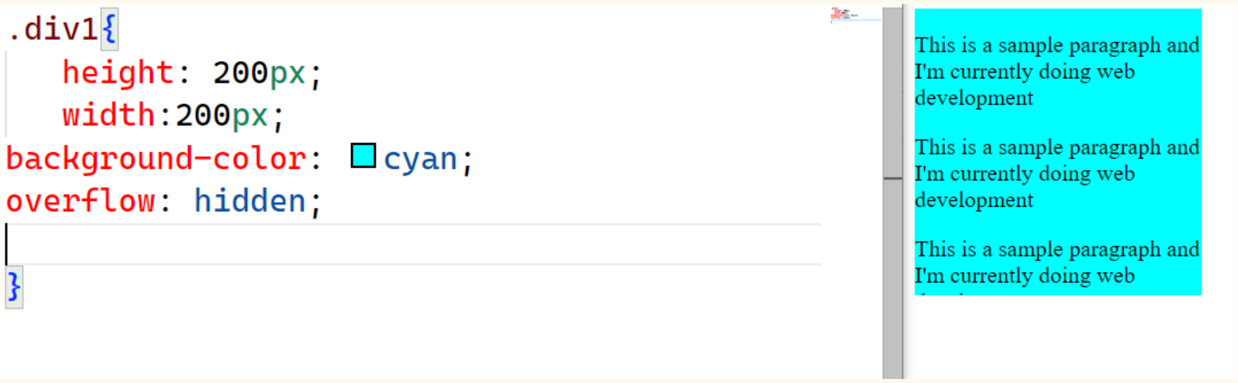
This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

2. Hidden



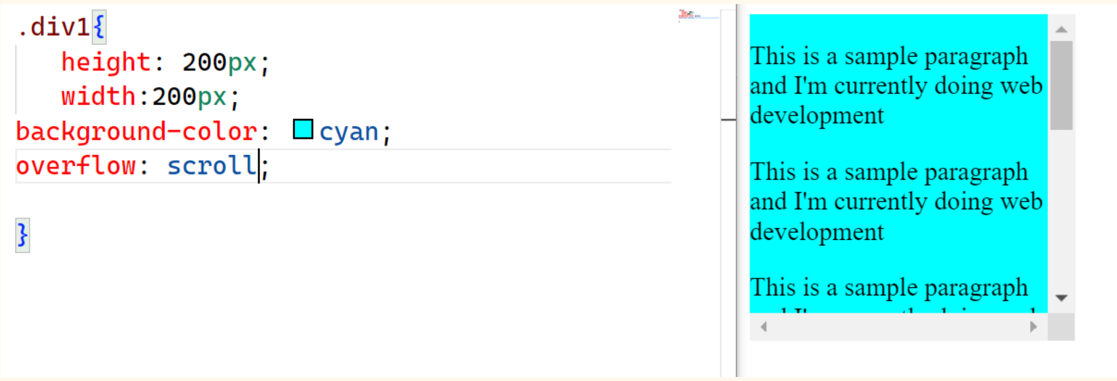
```
.div1{
    height: 200px;
    width:200px;
background-color: 🟦 cyan;
overflow: hidden;

}
```

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web

3. Scroll



```
.div1{
    height: 200px;
    width:200px;
background-color: 🟦 cyan;
overflow: scroll;

}
```

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph and I'm currently doing web development

This is a sample paragraph