1.How are inline and block elements different from each other?

Answers:

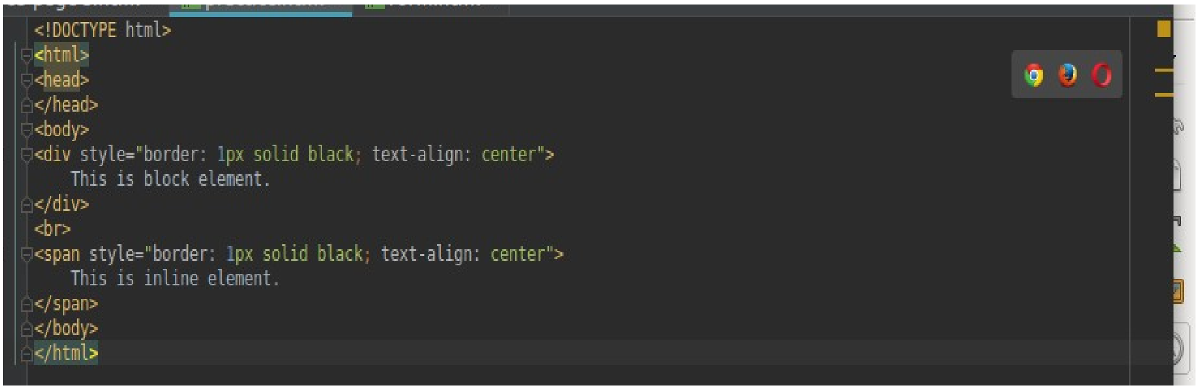
Html elements are divided into two categories inline and block elements:

Inline elements-

- Inline elements appear inside the body of an html page and can contain data or any other inline element.
- Inline elements do not begin on new lines by default.
- Inline elements create structures that are smaller as compared to block elements.
- Example of inline elements are <abbr>, <strong>, <span>etc.

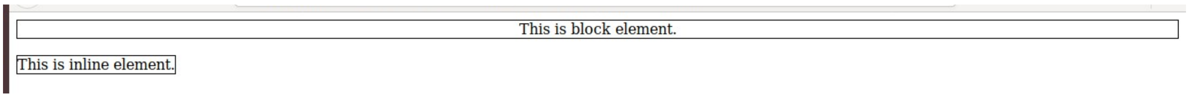
Block elements-

- block elements also appear inside the body of an html page and can contain another block element or inline element.
- Block elements begin on new lines by default.
- Block elements create structures that are larger as compared to inline elements.
- Example of inline elements are <p>, <h1>, <h2>, <div> etc.



```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<div style="border: 1px solid black; text-align: center">
    This is block element.
</div>
<br>
<span style="border: 1px solid black; text-align: center">
    This is inline element.
</span>
</body>
</html>
```

Output:



```
                         This is block element.
This is inline element.
```

2.Explain the difference between visibility:hidden and display:none

Answer:

- Visibility: hidden- Visibility: hidden is used to hide an element but the hidden element occupies some space as before and this also affects the layout of the webpage.

- Display: none- Display: none is also used to hide an element but the hidden element does not take any space and also does not affect the webpage layout.

3. Explain the clear and float properties.

Answer:

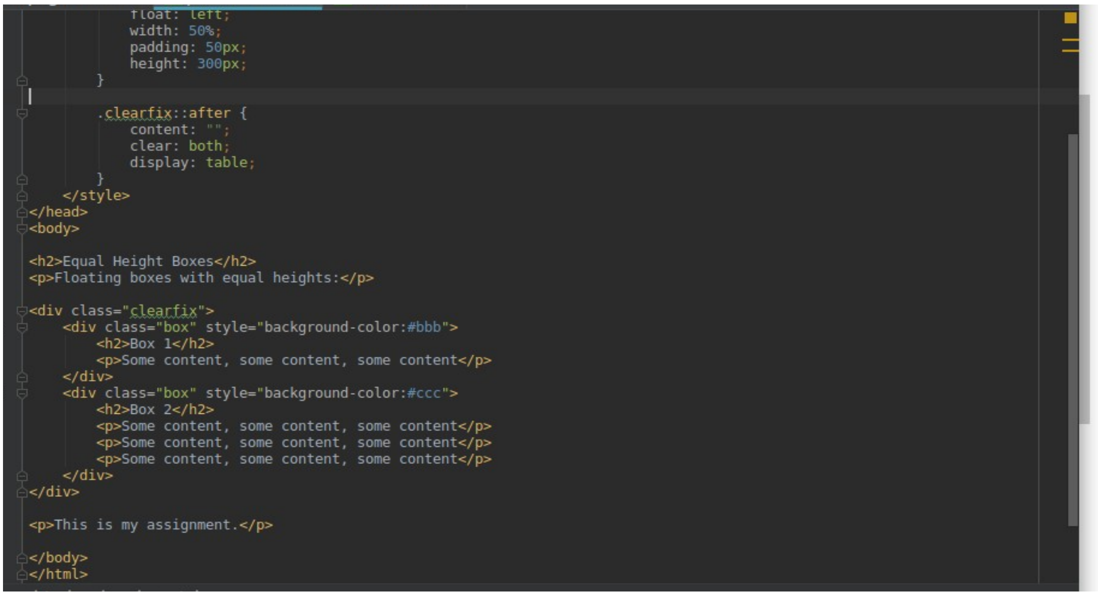
Float- Float property is used for positioning and formatting content. The basic use of the float property is to wrap text around images.

- The float property can have one of the following values: left, right, none and inherit.

Clear-Clear property is closely related to the float property. It specifies which elements can float beside the cleared element and on which side.

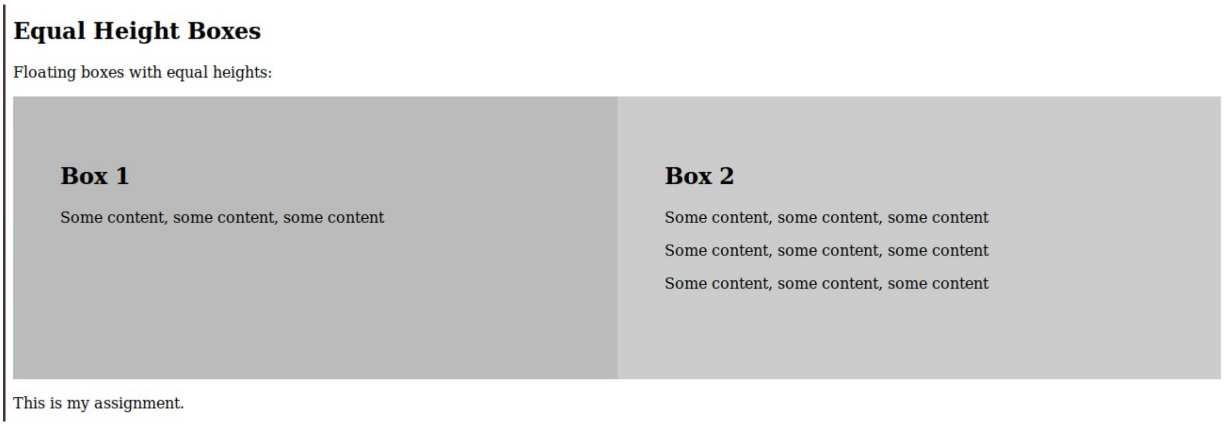
- The clear property can have one of the following values: left, right, both, none and inherit.

Floats:



```
                float: left;
                width: 50%;
                padding: 50px;
                height: 300px;
            }

            .clearfix::after {
                content: "";
                clear: both;
                display: table;
            }
    </style>
</head>
<body>

<h2>Equal Height Boxes</h2>
<p>Floating boxes with equal heights:</p>

<div class="clearfix">
    <div class="box" style="background-color:#bbb">
        <h2>Box 1</h2>
        <p>Some content, some content, some content</p>
    </div>
    <div class="box" style="background-color:#ccc">
        <h2>Box 2</h2>
        <p>Some content, some content, some content</p>
        <p>Some content, some content, some content</p>
        <p>Some content, some content, some content</p>
    </div>
</div>

<p>This is my assignment.</p>

</body>
</html>
```

Output:



Equal Height Boxes

Floating boxes with equal heights:

Box 1

Some content, some content, some content

Box 2

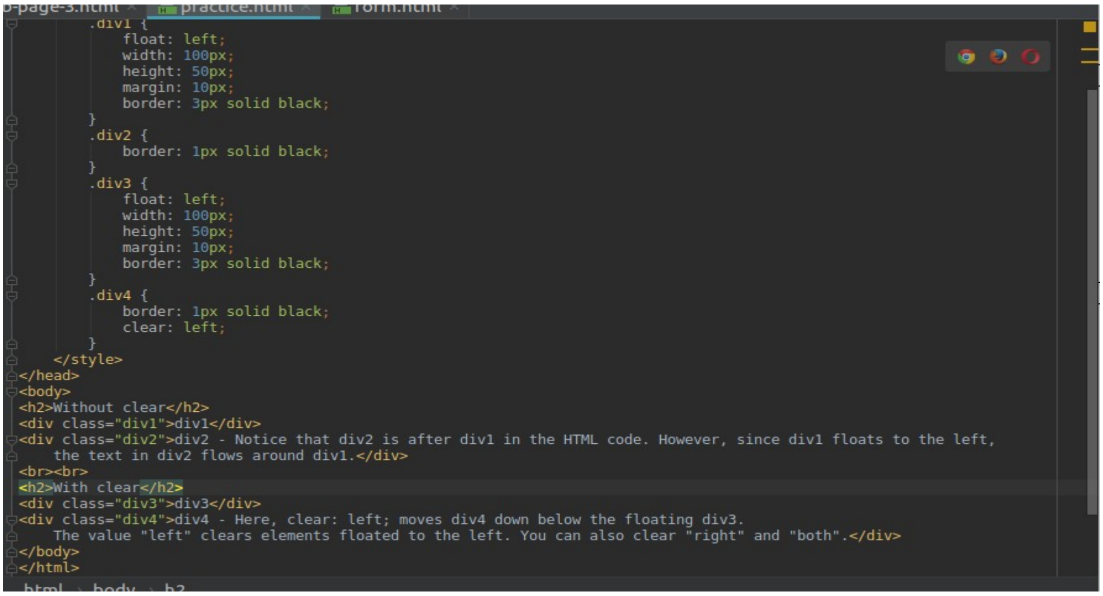
Some content, some content, some content

Some content, some content, some content

Some content, some content, some content

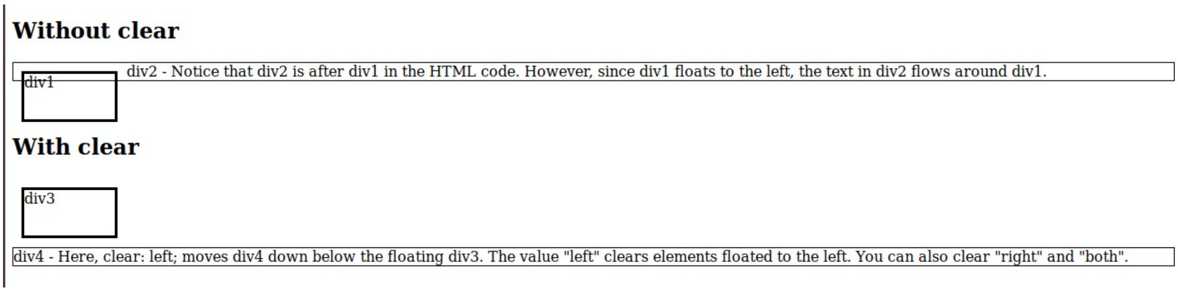
This is my assignment.

Clear:



```
        .div1 {
            float: left;
            width: 100px;
            height: 50px;
            margin: 10px;
            border: 3px solid black;
        }
        .div2 {
            border: 1px solid black;
        }
        .div3 {
            float: left;
            width: 100px;
            height: 50px;
            margin: 10px;
            border: 3px solid black;
        }
        .div4 {
            border: 1px solid black;
            clear: left;
        }
    </style>
</head>
<body>
<h2>Without clear</h2>
<div class="div1">div1</div>
<div class="div2">div2 - Notice that div2 is after div1 in the HTML code. However, since div1 floats to the left,
    the text in div2 flows around div1.</div>
<br><br>
<h2>With clear</h2>
<div class="div3">div3</div>
<div class="div4">div4 - Here, clear: left; moves div4 down below the floating div3.
    The value "left" clears elements floated to the left. You can also clear "right" and "both".</div>
</body>
</html>
```

Output:



Without clear

div1

div2 - Notice that div2 is after div1 in the HTML code. However, since div1 floats to the left, the text in div2 flows around div1.

With clear

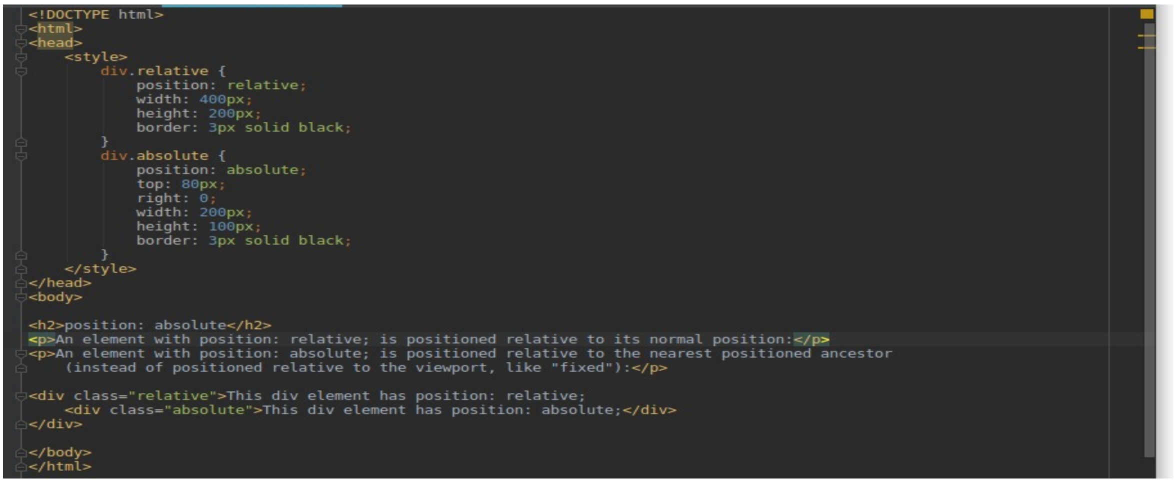
div3

div4 - Here, clear: left; moves div4 down below the floating div3. The value "left" clears elements floated to the left. You can also clear "right" and "both".

4. explain difference between absolute, relative,fixed and static.

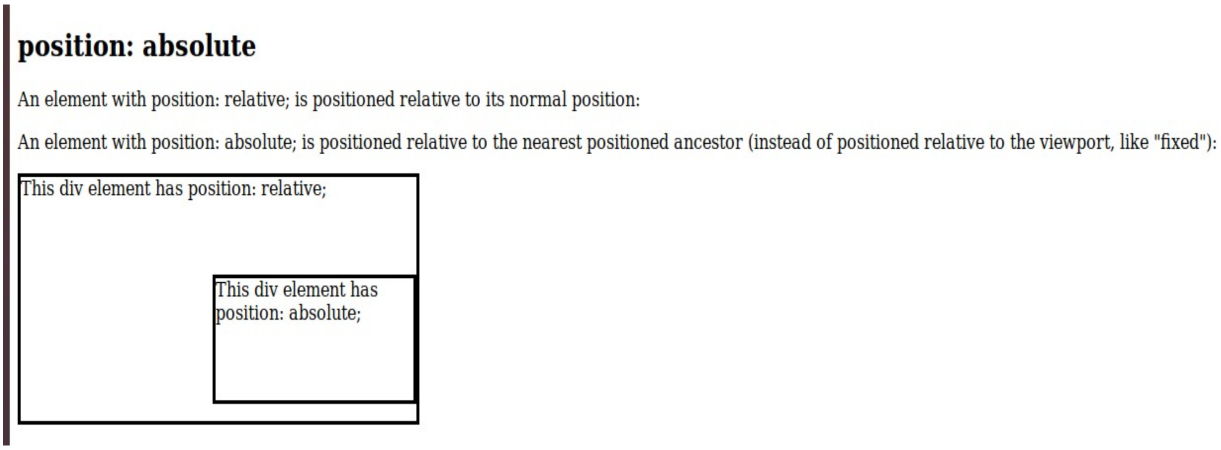
Answers:

Relative and Absolute:



```
<!DOCTYPE html>
<html>
<head>
    <style>
        div.relative {
            position: relative;
            width: 400px;
            height: 200px;
            border: 3px solid black;
        }
        div.absolute {
            position: absolute;
            top: 80px;
            right: 0;
            width: 200px;
            height: 100px;
            border: 3px solid black;
        }
    </style>
</head>
<body>

<h2>position: absolute</h2>
<p>An element with position: relative; is positioned relative to its normal position:</p>
<p>An element with position: absolute; is positioned relative to the nearest positioned ancestor
    (instead of positioned relative to the viewport, like "fixed"):</p>

<div class="relative">This div element has position: relative;
    <div class="absolute">This div element has position: absolute;</div>
</div>

</body>
</html>
```

Output:



position: absolute

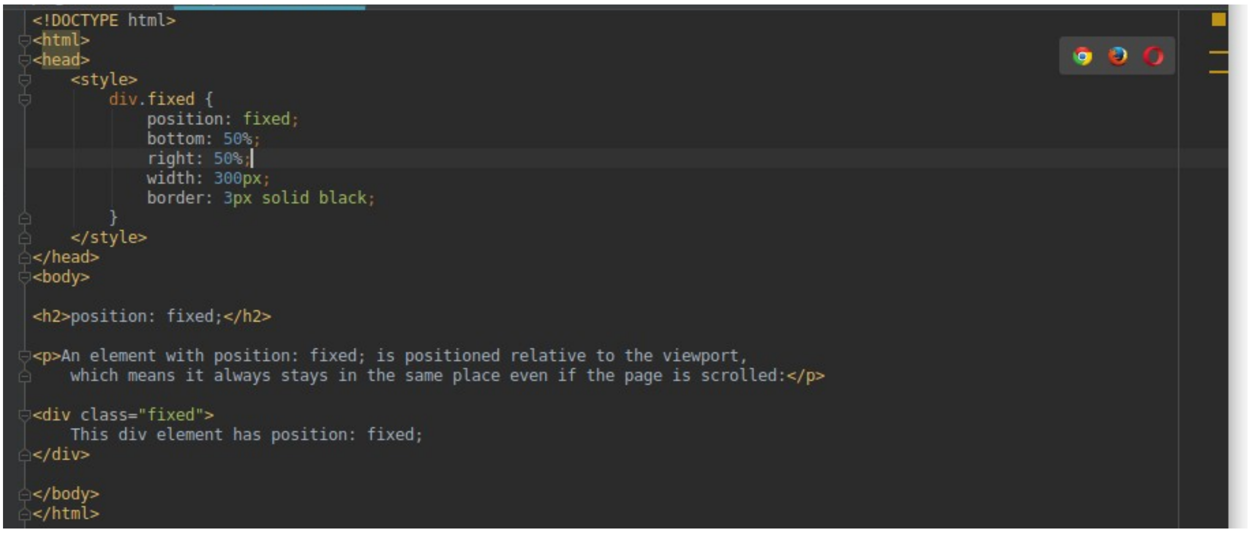
An element with position: relative; is positioned relative to its normal position:

An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like "fixed"):

This div element has position: relative;

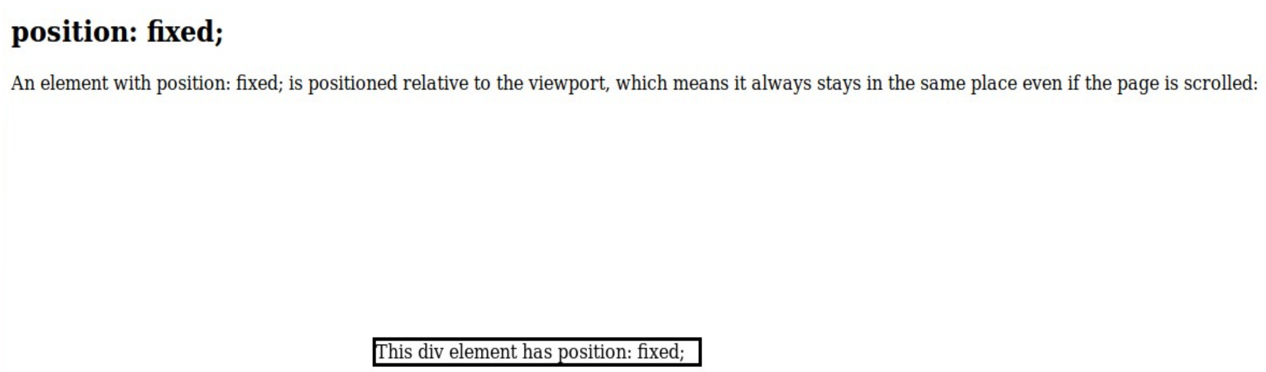
This div element has position: absolute;

Fixed:



```
<!DOCTYPE html>
<html>
<head>
    <style>
        div.fixed {
            position: fixed;
            bottom: 50%;
            right: 50%;
            width: 300px;
            border: 3px solid black;
        }
    </style>
</head>
<body>

<h2>position: fixed;</h2>

<p>An element with position: fixed; is positioned relative to the viewport,
    which means it always stays in the same place even if the page is scrolled:</p>

<div class="fixed">
    This div element has position: fixed;
</div>

</body>
</html>
```

Output:

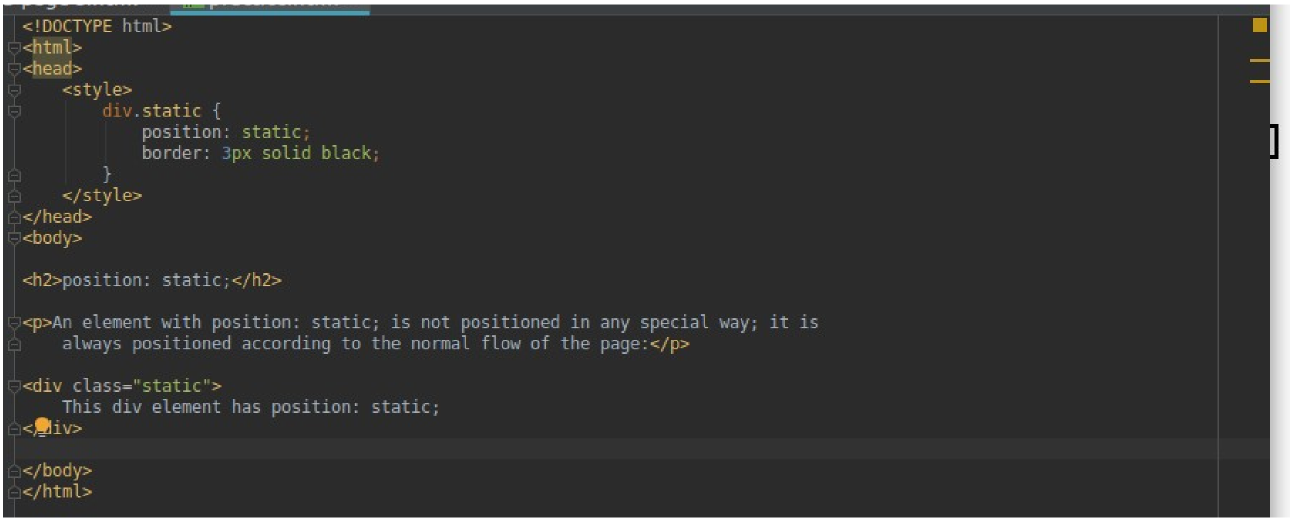


position: fixed;

An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled:

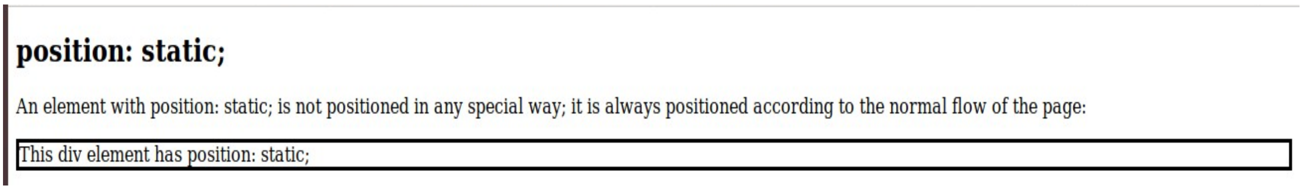
This div element has position: fixed;

Static:



```html
<!DOCTYPE html>
<html>
<head>
    <style>
        div.static {
            position: static;
            border: 3px solid black;
        }
    </style>
</head>
<body>

<h2>position: static;</h2>

<p>An element with position: static; is not positioned in any special way; it is
    always positioned according to the normal flow of the page:</p>

<div class="static">
    This div element has position: static;
</div>

</body>
</html>
```

Output:

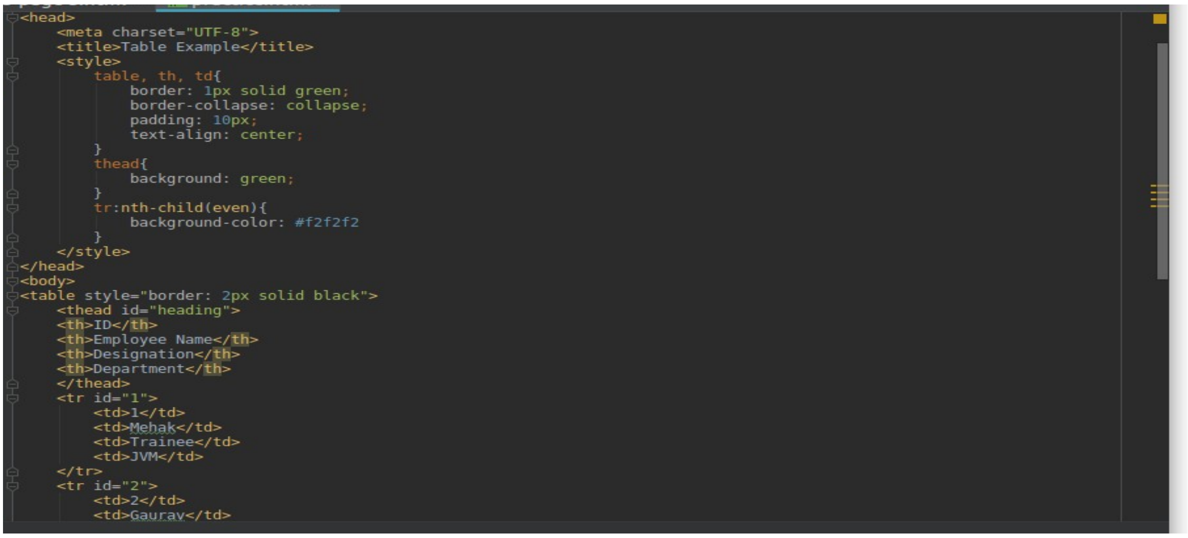


position: static;

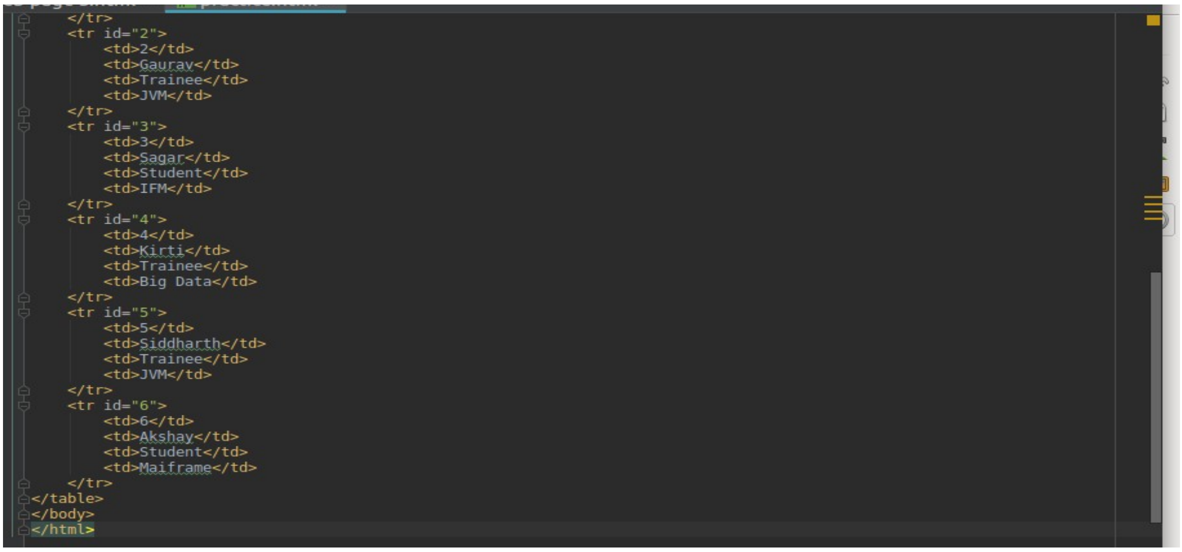
An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page:

This div element has position: static;

5. Write the HTML code to create a table in which there are 4 columns(ID , Employee Name, Designation, Department) and at least 6 rows. Also do some styling to it.

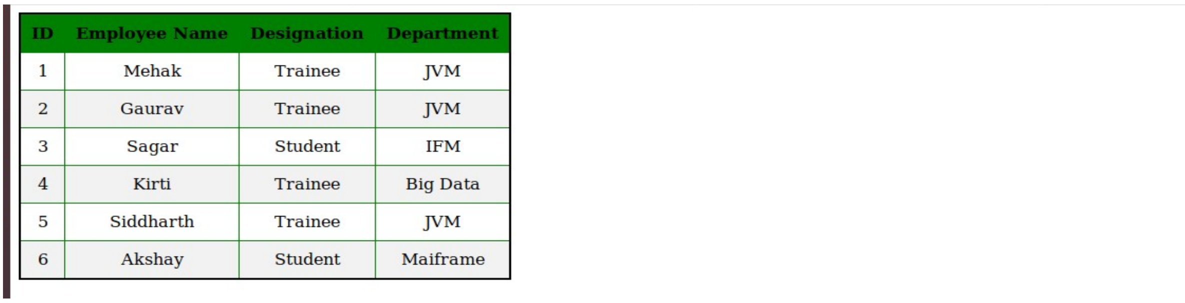


```html
<head>
    <meta charset="UTF-8">
    <title>Table Example</title>
    <style>
        table, th, td{
            border: 1px solid green;
            border-collapse: collapse;
            padding: 10px;
            text-align: center;
        }
        thead{
            background: green;
        }
        tr:nth-child(even){
            background-color: #f2f2f2
        }
    </style>
</head>
<body>
<table style="border: 2px solid black">
    <thead id="heading">
    <th>ID</th>
    <th>Employee Name</th>
    <th>Designation</th>
    <th>Department</th>
    </thead>
    <tr id="1">
        <td>1</td>
        <td>Mehak</td>
        <td>Trainee</td>
        <td>JVM</td>
    </tr>
    <tr id="2">
        <td>2</td>
        <td>Gaurav</td>
```



```
    </tr>
    <tr id="2">
        <td>2</td>
        <td>Gaurav</td>
        <td>Trainee</td>
        <td>JVM</td>
    </tr>
    <tr id="3">
        <td>3</td>
        <td>Sagar</td>
        <td>Student</td>
        <td>IFM</td>
    </tr>
    <tr id="4">
        <td>4</td>
        <td>Kirti</td>
        <td>Trainee</td>
        <td>Big Data</td>
    </tr>
    <tr id="5">
        <td>5</td>
        <td>Siddharth</td>
        <td>Trainee</td>
        <td>JVM</td>
    </tr>
    <tr id="6">
        <td>6</td>
        <td>Akshay</td>
        <td>Student</td>
        <td>Maiframe</td>
    </tr>
</table>
</body>
</html>
```

Output:



ID	Employee Name	Designation	Department
1	Mehak	Trainee	JVM
2	Gaurav	Trainee	JVM
3	Sagar	Student	IFM
4	Kirti	Trainee	Big Data
5	Siddharth	Trainee	JVM
6	Akshay	Student	Maiframe

6. Why do we use meta tags?

Answer:

Meta tags are used to provide additional information about the webpage.

Meta tags can be used in the following ways:

Specifying Keywords: Meta tags can be used for specifying important keywords that are later used by the search engines while indexing data during searching.

Ex- &lt;meta name = “keywords” content = “HTML, CSS, XML”/&gt;

Document description: Meta tags provide description about the document which is also used by search engines while indexing.

Ex- &lt;meta name = “description” content = “HTML and CSS assignment”/&gt;

Specifying author: You can use meta tags to set author name in a web page.

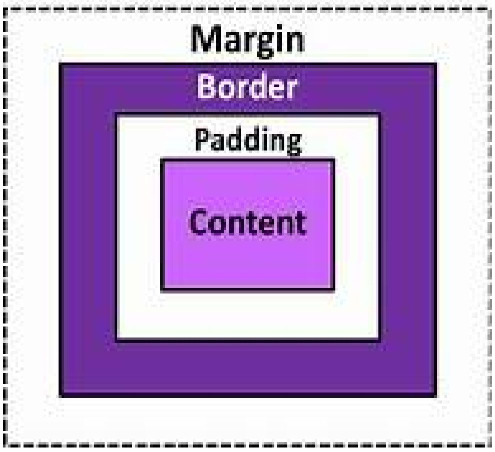
Ex- &lt;meta name = “author” content = “Mehak Adlakha” /&gt;

7. Explain box model.

Answer:

All html tags can be viewed as boxes. But in css the term box model refers to design and layout.

The basic idea behind the box model is of a box that wraps all the html elements around it. It consists of margins, borders, padding and the actual content.



- Content - The content of the box contains the actual data that is the text and images.
- Padding - Clears an area around the content. It is transparent.
- Border - A border that goes around the padding and content.
- Margin - Clears an area outside the border. It is also transparent.

8. What are the different types of CSS Selectors?

Answer:

- .[class_name]- The dot selector selects all the elements with the class name.

- #[id]- The hashtag selector selects all the elements with the specified id.

- *- The asterisk selector selects all the elements.

- Element- The element selector selects all the elements of the specified tag.

- element1,element2- It selects all the elements of element 1 and element 2.

- element1 element2- Selects all the element2 elements present in element1.

9. Define Doctype.

Answer:

- <DOCTYPE!> though it seems, is not an HTML tag, rather it is an instruction to inform the web browser what version of HTML, the file is written in.
- <DOCTYPE! html> is declaration for HTML 5, which should come before the <html> tag.
- It is case insensitive.

10. Explain 5 HTML5 semantic tags.

Answer:
<article> - Defines an article in the document

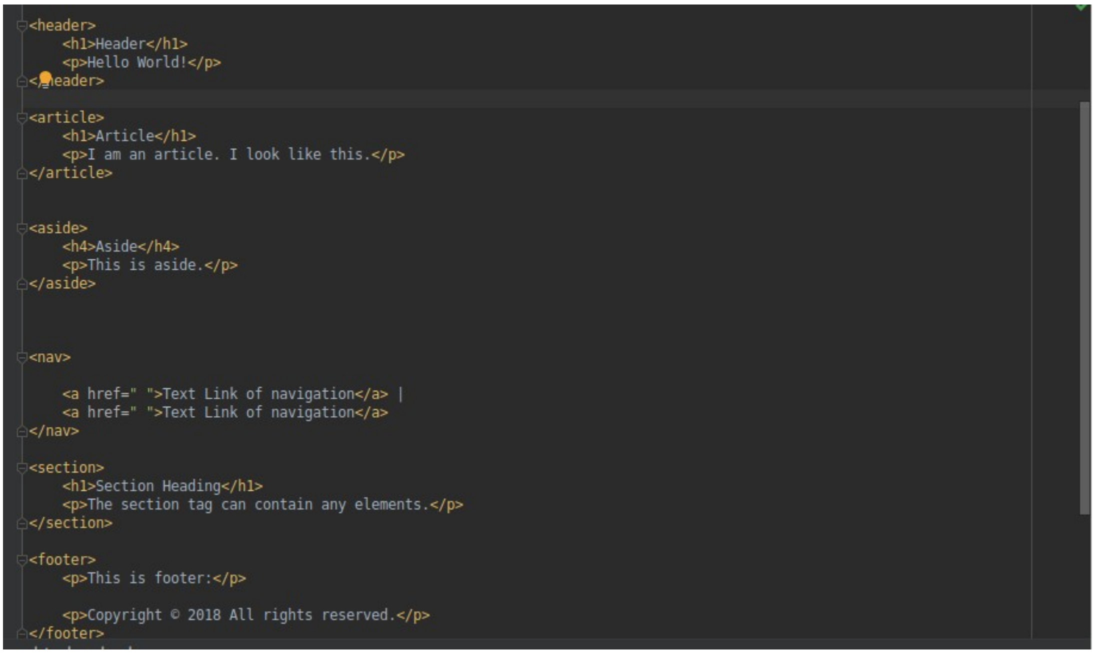
<aside> - Defines content aside from the page content like sidebar.

<footer> - Defines a footer for the document or a section

<header> - Defines a header for the document or a section

<nav> - Defines navigation links in the document

<section> - Defines a section in the document



```
<header>
    <h1>Header</h1>
    <p>Hello World!</p>
</header>

<article>
    <h1>Article</h1>
    <p>I am an article. I look like this.</p>
</article>


<aside>
    <h4>Aside</h4>
    <p>This is aside.</p>
</aside>



<nav>

    <a href=" ">Text Link of navigation</a> |
    <a href=" ">Text Link of navigation</a>
</nav>

<section>
    <h1>Section Heading</h1>
    <p>The section tag can contain any elements.</p>
</section>

<footer>
    <p>This is footer:</p>

    <p>Copyright © 2018 All rights reserved.</p>
</footer>
```

Output:

Header

Hello World!

Article

I am an article. I look like this.

Aside

This is aside.

Text Link of navigation | Text Link of navigation

Section Heading

The section tag can contain any elements.

This is footer: