Contents

[CSS 1](#_Toc69199818)

[**CSS BOX MODEL** 1](#_Toc69199819)

[**BLOCK AND INLINE ELEMENTS** 1](#_Toc69199820)

[SOME IMPORTANT CSS PROPERTIES 1](#_Toc69199821)

[**POSITION** 1](#_Toc69199822)

[**TEXT FORMATING** 1](#_Toc69199823)

[**WIDTH AND HEIGHT** 2](#_Toc69199824)

[**LIST PROPERTY** 2](#_Toc69199825)

[**OVERFLOW** 2](#_Toc69199826)

[**LINE HEIGHT** 2](#_Toc69199827)

[**MEDIA QUERIES** 2](#_Toc69199828)

# CSS

## **CSS BOX MODEL**

|  |  |
| --- | --- |
|  | * Every HTML element in follows the box model. Means it will have always have content area, padding, border and margin associated with them. They are transparent by default * The padding and border are not visible by default * The browser adds a default margin between all the elements in the HTML |

### **BLOCK AND INLINE ELEMENTS**

* The HTML elements are primarily divided on 2 types a Block and inline elements

**BLOCK ELEMENTS**

1. The block elements are the main structural element of the HTML [div,p, h1]
2. By default - It is painted from left to right on the browser and never shared the space with other block elements i.e. cannot be stacked side by side.

**INLINE ELEMENTS**

1. Span , image tags etc are inline elements – which when added to HTML – It shares the space i.e it is stacked side by side.
2. Inline elements reposition themselves when we change the browser window size

## SOME IMPORTANT CSS PROPERTIES

### **POSITION**

|  |  |
| --- | --- |
| Absolute position is always the absolute position with respect to browser window.  The position of the elements adjust itself based on the window size | Example  p {  **position: absolute;**  top: 200px;  right: 100px;  } |
| The fixed position will fix the elements on fixed position |  |

### **TEXT FORMATING**

* **font-family**:Verdana,Arial,Arial-Black : We specify the font family which we want to load . We specify multiple font families as a fallback of the previous one.
* **font**-weight:lighter|normal|bold|bolder
* **text**-decoration:underline|line-through|overline 🡪underline and overline will add underline and overline respectively . line-through will strike-out the line.

### **WIDTH AND HEIGHT**

* ***The height and width can only be set for block element like div, p etc***.
* The height and width can be set in **px [fixed] or %[Percentage of browser screen- Good for responsive design]**
* **“auto” – This is the default value of width and height of not set explicitly**
* min-width and max-width is used to set the minimum and maximum value of the element

### **LIST PROPERTY**

|  |  |
| --- | --- |
| **list-style-type** | This is to specify how the list bullets look like. Like disc, circle,square etc. e.g. ul { list-style-type: square; } |
| **list-style-image** | To use image as list item marker.e.g 🡪 **ul {list-item-image:url(‘image.png’)}** |
| **list-style-position** | It specifies the position of the list item. ul { list-style-position: outside|inside;} |
| **list-style** |  |

### **OVERFLOW**

This property is used to define – what will happen if the content overflows the element’s box. It defines to whether to clip the content or add a scrollbar. This property works only from block element

e.g. p{overflow: visible(default| scroll | hidden|auto)}

|  |  |
| --- | --- |
| **visible** | It’s a default property. It will show the content even if the content outside the dimension of wrapper element |
| **scroll** | It will show the scrollbar every time |
| **hidden** | The content will not show up if it falls outside the dimension of wrapper container |
| **auto** | It will show the scrollbar only if the content overflow from the size of container |

### **LINE HEIGHT**

This property is

***INHERITACE IN CSS***

* Usually All the child elements inherits the CSS property of the parent element.
* Some of the child element does not follow the inheritance like link tag(<a>).

### **MEDIA QUERIES**

|  |  |
| --- | --- |
| @media (max-width:600px){  #box{background-color:red}  }  @media (min-width:600px){  #box{background-color:red}  } | 1. max-width means the css will be applied when we screen size is below that pixel size 2. min-width means the css will be applied when we screen size is beyond/above that pixel size |

**USE CASES: CREATING A VERTICAL MENU**

|  |  |
| --- | --- |
| **CREATE A LIST ITEM IN HTML**  <ul>  <li><a href="#">Home</a></li>  <li><a href="#">Products</a></li>  <li><a href="#">Contact</a></li>  </ul> |  |
| **MANIPULATING LIST ITEMS**   * 1. Remove the bullets from list items   2. Get rid of default margin and padding | ul {  list-style: none;  margin  } |
| **MANIPULATING THE A TAG**   1. Remove the underline and add font-family to it. 2. Add the background color and font color.   Note – while adding a background color to hyperlink – it will fill the color only the hyperlink as it’s an inline element. So we have convert it to a block element   1. Since when it converted into block element it will occupy the 100% browser width so we need to specify the width to limit the width 2. Add the padding ,margin and center align the text 3. To appear it as button – we will give border with a specific width 4. Add a height to the link and center align the text vertically(using line height | ul li a {  text-decoration: none;  font-family: Verdana, Geneva, Tahoma, sans-serif;  background-color: grey;  display: block;  margin: 15px;  padding: 5px;  width: 80px;  text-align: center;  border: red solid 1px;  border-radius: 1px;  height: 30px;  line-height: 25px;  } |
| **ADDING A HOVER EFFECT** | ul li a:**hover** {  background-color: blue;  color: #fff;  } |

**USE CASES: CREATING A VERTICAL MENU**

|  |  |
| --- | --- |
| **CREATE A WRAPPER ELEMENT AND LIST ITEM** | <div id="nav">  <ul>  <li><a href="#">Home</a></li>  <li><a href="#">Products</a></li>  <li><a href="#">Contact</a></li>  </ul>  </div> |
| **ADD STYLING TO WRAPPER** | #nav {  background-color: green;  font-family: Verdana, Geneva, Tahoma, sans-serif;  height: 25px;  } |
| **ADD STYLING TO UL** | #nav ul {  list-style: none;  margin: 0px;  padding: 0px;  } |
| **ADD STYLING TO LI** | #nav li {  display: inline;  padding: 15px;  vertical-align: middle;  } |
| **ADDING HOVER** | #nav ul li:hover {  background-color: white;  } |