

## Media queries in CSS

To make the responsive we write mediaqueries, In these queries we use conditions based on screen size in pixel

<pre>To assign the format if the screen size&lt;480px @media screen and (max-width:480px){     div,h1{         background-color: rgb(0, 255, 183);         border:2px solid green;         border-radius:5px ;         font-size: 50px;         width:60%     }     h1{         background-color: rgb(255, 0, 123);         color:white;     } }</pre>	<pre>To assign the format if the screen size&gt;=481px and &lt;=600px  @media screen and (min-width:481px) and (max-width:600px){     div,h1{         background-color: rgb(115, 0, 255);         color:white;         border:2px solid palevioletred;         border-radius:5px ;         font-size: 30px;         width:60%     }     h1{         background-color: yellow;     } }</pre>
--	---

## Animation effect

To give animation effect we need to use keyframes

<pre>@keyframes animation1 {     from{         margin-left:100px;         background-color: pink;         width:400px;     }     to{         margin-left:100px;         margin-top: 50px;         background-color: rgb(211, 192, 255);         width:300px;     } }</pre>	<pre>@keyframes animation2 {     10%{         margin-left:100px;         background-color: pink;         width:400px;     }     20%{         margin-left:100px;         margin-top: 50px;         background-color: rgb(211, 192, 255);         width:300px;     }     40%{margin-left:200px;         background-color: rgb(13, 215, 20);         width:400px;     }     50%{         margin-left:100px;         margin-top:200px;         background-color: rgb(13, 215, 20);         width:300px;     } }</pre>
--	---

	<pre> 80%{   margin-left:50px;   margin-top:50px;   background-color: rgb(18, 169, 199);   width:200px; } 100%{   margin-left:500px;   margin-top:100px;   background-color: rgb(222, 105, 9);   width:200px; } } </pre>
--	--

To make div behave like a container add property display:flex

flex container

The properties are divided into container property and item property

Container	<ol style="list-style-type: none"> <li>1. Flex-direction→ the direction in which the items will appear</li> <li>2. Flex-wrap→ it decides, that when the screen size reduces, then whether to wrap the items or not</li> <li>3. Flex-flow→ it is a shortcut for Flex-direction and Flex-wrap</li> <li>4. Justify-content→ It aligns the items horizontally</li> <li>5. Align-items→ It aligns the items vertically</li> <li>6. Align-content→ It is a shortcut for Justify-content and Align-items</li> </ol>
Flex items	<ol style="list-style-type: none"> <li>1. Order→ it displays the sequence in ascending order based on order property's value</li> <li>2. Flex-grow→ how fast the size of the item should grow, when the screen size is changing</li> <li>3. Flex-shrink---how fast the item should shrink, when the screen size is changing</li> <li>4. Flex-basis→ it specifies initial size of the item</li> <li>5. Flex→ it is shortcut for Flex-grow Flex-shrink flex-basis</li> <li>6. Align-self→ the item align itself, it overrides the container's alignment</li> </ol>

## JavaScript- It is a scripting language

### 2 types of Javascript

1. Internal javascript→ if the javascript code is written inside HTML page, in <script> tag, then it is called as internal javascript
2. External javascript--→ If we write separate .js file, and add it in HTML page using <script> tag, then it is called as external javascript

### Java VS Javascript

Java is a programming language, in which the code will get executed sequentially.	In Javascript the program is event driven, the execution of code is dependent on user action
Java is statically type, means the data type of variable is decided at compile time	In javascript type of variables is decided at run time, based on the value you have assigned to the variable
Java is Object oriented.	Javascript is object based.

### In Javascript

To display data in the browser window	<code>document.write("&lt;h1&gt;Hello World&lt;/h1&gt;");</code>
To display in console window	<code>console.log("In console window");</code>
To display popup with ok button	<code>Alert("Hello in alert")</code>
To display popup with ok and cancel button	<code>Confirm("in confirm box")</code> If user clicks on ok, then it returns true, if user clicks on cancel, then it returns false
To display popup with message and a text box	<code>Prompt("enter name")</code> If user clicks on ok then it returns value in the text box, otherwise it returns null

To handle click event, use onclick property on the button

The tags div, p, span table has innerHTML property

The textbox, radiobutton, checkbox has value property