CSS Media Queries

What is Media Query?

CSS Media query is a W3C recommendation and a CSS3 module which is used to adapt to conditions such as screen resolution (e.g. Smartphone screen vs. computer screen).

* The media query technique first used in CSS3.
* It became a W3C recommendation in June 2012.
* It is an extension of media dependent stylesheets used in different media types (i.e. screen and print) found in CSS2.
* The most commonly used media feature is "width".
* It uses the @media rule to include a block of CSS properties only if a certain condition is true.

Recommended Media features for Media queries:

Following is a list of media features recommended for media queries by W3C.

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Value** | **Min/Max** | **Description** |
| color | integer | yes | It specifies the number of bits per color component. |
| color-index | integer | yes | It specifies the number of entries in the color lookup table. |
| device-aspect-ratio | integer/integer | yes | It specifies the aspect ratio of the device. |
| device-height | length | yes | It specifies the height of the output device. |
| device-width | length | yes | It specifies the width of the output device. |
| grid | integer | no | It is true for a grid-based device. |
| height | length | yes | It specifies the height of the rendering surface. |
| monochrome | integer | yes | It specifies the number of bits per pixel in a monochrome frame buffer. |
| resolution | resolution ("dpi" or "dpcm") | yes | It specifies the resolution of the display screen. |
| scan | "progressive" or "interlaced" | no | It specifies scanning process of "tv" media types. |
| width | length | yes | It specifies the width of the rendering surface. |

What is Responsive Web Design?

The term Responsive Web Design was given by Ethan Marcotte. It facilitates you to use fluid grids, flexible images, and media queries to progressively enhance a web page for different viewing contexts i.e. Desktop, Smartphone, Tablet etc.

What screen resolutions do you use while taking screenshots?

Smartphone: 320px

Tablet: 768px

Netbook: 1024px

Desktop: 1600px

Let's take an example to see the simple use of media query:

This example specifies that if you resize your window less than 500px, the background color will be changed.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<meta** name="viewport" content="width=device-width, initial-scale=1.0"**/>**
5. **<style>**
6. body {
7. background-color:yellow;
8. }
10. @media only screen and (max-width: 500px) {
11. body {
12. background-color:green;
13. }
14. }
15. **</style>**
16. **</head>**
17. **<body>**
18. **<p>**If you resize the browser window and the width of this document is less than 500 pixels, the background-color is "green", otherwise it is "yellow"**</p>**
19. **</body>**
20. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-media-query1)

How to add a breakpoint?

Media query can be used to create a responsive webpage. The breakpoint is used on web pages where you want that certain parts of the design will behave differently on each side of the breakpoint.

**Let's take an example:**

Here we use a media query to add a breakpoint at 768px.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<meta** name="viewport" content="width=device-width, initial-scale=1.0"**>**
5. **<style>**
6. \* {
7. box-sizing: border-box;
8. }
9. .row:after {
10. content: "";
11. clear: both;
12. display: block;
13. }
14. [class\*="col-"] {
15. float: left;
16. padding: 15px;
17. }
18. html {
19. font-family: "Lucida Sans", sans-serif;
20. }
21. .header {
22. background-color: purple;
23. color: pink;
24. padding: 15px;
25. }
26. .menu ul {
27. list-style-type: none;
28. margin: 0;
29. padding: 0;
30. }
31. .menu li {
32. padding: 8px;
33. margin-bottom: 7px;
34. background-color :green;
35. color: #ffffff;
36. box-shadow: 0 1px 3px rgba(0,0,0,0.12), 0 1px 2px rgba(0,0,0,0.24);
37. }
38. .menu li:hover {
39. background-color: #0099cc;
40. }
41. .aside {
42. background-color: green;
43. padding: 15px;
44. color: #ffffff;
45. text-align: center;
46. font-size: 14px;
47. box-shadow: 0 1px 3px rgba(0,0,0,0.12), 0 1px 2px rgba(0,0,0,0.24);
48. }
49. .footer {
50. background-color: #0099cc;
51. color: #ffffff;
52. text-align: center;
53. font-size: 12px;
54. padding: 15px;
55. }
56. /\* For mobile phones: \*/
57. [class\*="col-"] {
58. width: 100%;
59. }
60. @media only screen and (min-width: 600px) {
61. /\* For tablets: \*/
62. .col-m-1 {width: 8.33%;}
63. .col-m-2 {width: 16.66%;}
64. .col-m-3 {width: 25%;}
65. .col-m-4 {width: 33.33%;}
66. .col-m-5 {width: 41.66%;}
67. .col-m-6 {width: 50%;}
68. .col-m-7 {width: 58.33%;}
69. .col-m-8 {width: 66.66%;}
70. .col-m-9 {width: 75%;}
71. .col-m-10 {width: 83.33%;}
72. .col-m-11 {width: 91.66%;}
73. .col-m-12 {width: 100%;}
74. }
75. @media only screen and (min-width: 768px) {
76. /\* For desktop: \*/
77. .col-1 {width: 8.33%;}
78. .col-2 {width: 16.66%;}
79. .col-3 {width: 25%;}
80. .col-4 {width: 33.33%;}
81. .col-5 {width: 41.66%;}
82. .col-6 {width: 50%;}
83. .col-7 {width: 58.33%;}
84. .col-8 {width: 66.66%;}
85. .col-9 {width: 75%;}
86. .col-10 {width: 83.33%;}
87. .col-11 {width: 91.66%;}
88. .col-12 {width: 100%;}
89. }
90. **</style>**
91. **</head>**
92. **<body>**
94. **<div** class="header"**>**
95. **<h1>**JavaTpoint**</h1>**
96. **</div>**
98. **<div** class="row"**>**
99. **<div** class="col-3 col-m-3 menu"**>**
100. **<ul>**
101. **<li>**C/C++**</li>**
102. **<li>**Java**</li>**
103. **<li>**SQL**</li>**
104. **<li>**Android**</li>**
105. **<li>**HTML**</li>**
106. **<li>**CSS**</li>**
107. **<li>**Cloud Computing**</li>**
108. **<li>**PHP**</li>**
109. **<li>**JSON**</li>**
110. **<li>**JQuery**</li>**
111. **<li>**MongoDB**</li>**
112. **<li>**Oracle**</li>**
114. **</ul>**
115. **</div>**
117. **<div** class="col-6 col-m-9"**>**
118. **<h1>**About JavaTpoint**</h1>**
119. **<p>**JavaTpoint is written and developed that students may learn computer science related technologies easily.**</p>**
120. **</div>**
122. **<div** class="col-3 col-m-12"**>**
123. **<div** class="aside"**>**
124. **<h2>**What is JavaTpoint?**</h2>**
125. **<p>**JavaTpoint is the No.1 Java training institute in Noida, Delhi, Gurgaon, Ghaziabad and Faridabad.
126. You will get practical training on Java by our Java expert who have 7+ year industrial experience.**</p>**
127. **<h2>**Why JavaTpoint?**</h2>**
128. **<p>**Life Time Validity, Training by Java Professionals, Problem Solving Team, Project Development,
129. Small Batches to focus on each student
130. **</p>**
131. **<h2>**How to reach?**</h2>**
132. **<p>**JavaTpoint is located in Noida (Gautam Budhh Nagar). The full address is G-13 Second
133. Floor Sector 3 (Near Sector-16 Metro Station)
134. Noida(U.P)**</p>**
135. **</div>**
136. **</div>**
138. **</div>**
140. **<div** class="footer"**>**
141. **<p>**Resize the browser window to see how the content respond to the resizing.**</p>**
142. **</div>**
144. **</body>**
145. **</html>**

CSS Transforms

CSS3 supports transform property. This transform property facilitates you to translate, rotate, scale, and skews elements.

Transformation is an effect that is used to change shape, size and position.

There are two type of transformation i.e. 2D and 3D transformation supported in CSS3.

CSS 2D Transforms

The CSS 2D transforms are used to re-change the structure of the element as translate, rotate, scale and skew etc.

Following is a list of 2D transforms methods:

* **translate(x,y):** It is used to transform the element along X-axis and Y-axis.
* **translateX(n):** It is used to transform the element along X-axis.
* **translateY(n):** It is used to transform the element along Y-axis.
* **rotate():** It is used to rotate the element on the basis of an angle.
* **scale(x,y):** It is used to change the width and height of an element.
* **scaleX(n):** It is used to change the width of an element.
* **scaleY(n):** It is used to change the height of an element.
* **skewX():** It specifies the skew transforms along with X-axis.
* **skewY():**It specifies the skew transforms along with Y-axis.
* **matrix():** It specifies matrix transforms.

# CSS 3D Transforms

The CSS 3D transforms facilitates you to move an element to X-axis, Y-axis and Z-axis. Following is a list of 3D transforms methods:

|  |  |
| --- | --- |
| **Function** | **Description** |
| matrix3D(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n) | It specifies a 3D transformation, using a 4x4 matrix of 16 values. |
| translate3D(x,y,z) | It specifies a 3D translation. |
| translateX(x) | It specifies 3D translation, using only the value for the X-axis. |
| translateY(y) | It specifies 3D translation, using only the value for the Y-axis. |
| translateZ(z) | It specifies 3D translation, using only the value for the Z-axis. |
| scale3D(x,y,z) | It specifies 3D scale transformation |
| scaleX(x) | It specifies 3D scale transformation by giving a value for the X-axis. |
| scaley(y) | It specifies 3D scale transformation by giving a value for the Y-axis. |
| scaleZ(z) | It specifies 3D scale transformation by giving a value for the Z-axis. |
| rotate3D(X,Y,Z,angle) | It specifies 3D rotation along with X-axis, Y-axis and Z-axis. |
| rotateX(angle) | It specifies 3D rotation along with X-axis. |
| rotateY(angle) | It specifies 3D rotation along with Y-axis. |
| rotateZ(angle) | It specifies 3D rotation along with Z-axis. |
| perspective(n) | It specifies a perspective view for a 3D transformed element. |

CSS Aural Media/ Style Sheets

CSS Aural Media or CSS Style Sheets are used to attach specific sound style features to specific document elements. It uses speech synthesis and sound effect to facilitate users to listen to information instead of reading them. So, it is very useful for visually impaired people.

Aural media can be used in following:

* Used by blind or visually impaired people
* Help users learning to read and right pronunciation
* Training
* Help users who have reading problems
* Facilitates web access in vehicles
* Home entertainment
* Used by print-impaired communities
* Medical documentation
* Industrial documentation

Aural Media Properties

Following is a list of aural media properties:

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
| **property** | **explanation** | **value** |
| azimuth | it is set where the sound should come from. | angle  left-side  far-left  left  center-left  center  center-right  right  far-right  right-side  behind  leftwardsrightwards |
| cue | it is used to set the cue properties in one declaration. | cue-beforecue-after |
| cue-after | it is used to specify a sound to be played after speaking an element's content. | none  url |
| cue-before | it is used to specify a sound to be played before speaking an element's content. | noneurl |
| elevation | it is set where the sound should come from. | angle  below  level  above  higherlower |
| pause | it is used to set the pause properties in one declaration. | pause-beforepause-after |
| pause-after | it is used to specify a pause after speaking an element's content. | time  % |
| pause-before | it is used to specify a pause before speaking an element's content. | time  % |
| pitch | it is used to specify the pitch of the speaking voice. | frequency  x-low  low  medium  high  x-high |
| pitch-range | it is used to specify the variation of the speaking voice. it specifies whether he speaking voice is monotone voice or animated voice. | number |
| play-during | it specifies a sound to be played while speaking an element's content. | auto  none  url  mix  repeat |
| richness | it is used to specifies the sound quality of the speaking voice. it specifies that the voice is rich or thin. | number |
| speak | it specifies whether content will render aurally. | normal  nonespell-out |
| speak-header | it is used to specify how to handle table headers. it also specifies whether the headers be spoken before every cell, or only before a cell with a different header than the previous cell. | always  once |
| speak-numeral | it specifies how to speak numbers. | digits  continuous |
| speak-punctuation | it is used to specify how to speak punctuation characters. | nonecode |
| speech-rate | it specifies the speed of the speaking. | number  x-slow  slow  medium  fast  x-fast  fasterslower |
| stress | it is used to specify the "stress" in the speaking voice. | number |
| voice-family | it specifies the voice family of the speaking voice. | specific-voice  generic-voice |
| volume | it specifies the volume of the speaking voice. | number  %  silent  x-soft  soft  medium  loud  x-loud |

CSS 3D Transforms

The CSS 3D transforms facilitates you to move an element to X-axis, Y-axis and Z-axis. Following is a list of 3D transforms methods:

|  |  |
| --- | --- |
| **Function** | **Description** |
| matrix3D(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n) | It specifies a 3D transformation, using a 4x4 matrix of 16 values. |
| translate3D(x,y,z) | It specifies a 3D translation. |
| translateX(x) | It specifies 3D translation, using only the value for the X-axis. |
| translateY(y) | It specifies 3D translation, using only the value for the Y-axis. |
| translateZ(z) | It specifies 3D translation, using only the value for the Z-axis. |
| scale3D(x,y,z) | It specifies 3D scale transformation |
| scaleX(x) | It specifies 3D scale transformation by giving a value for the X-axis. |
| scaley(y) | It specifies 3D scale transformation by giving a value for the Y-axis. |
| scaleZ(z) | It specifies 3D scale transformation by giving a value for the Z-axis. |
| rotate3D(X,Y,Z,angle) | It specifies 3D rotation along with X-axis, Y-axis and Z-axis. |
| rotateX(angle) | It specifies 3D rotation along with X-axis. |
| rotateY(angle) | It specifies 3D rotation along with Y-axis. |
| rotateZ(angle) | It specifies 3D rotation along with Z-axis. |
| perspective(n) | It specifies a perspective view for a 3D transformed element. |

Supporting Browsers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Property** | **chrome browser Chrome** | **ie browser IE** | **firefox browser Firefox** | **opera browser Opera** | **safari browser Safari** |
| transform | 36.012.0 -webkit- | 10.0 | 16.0  10.0 -moz- | 23.015.0 -webkit- | 9.04.0 -webkit- |
| transform-origin  (three-value syntax) | 36.012.0 -webkit- | 10.0 | 16.010.0 -moz- | 23.015.0 -webkit- | 9.04.0 -webkit- |
| transform-style | 36.012.0 -webkit- | 11.0 | 16.010.0 -moz- | 23.015.0 -webkit- | 9.04.0 -webkit- |
| perspective | 36.012.0 -webkit- | 10.0 | 16.010.0 -moz- | 23.015.0 -webkit- | 9.04.0 -webkit- |
| perspective-origin | 36.012.0 -webkit- | 10.0 | 16.010.0 -moz- | 23.015.0 -webkit- | 9.04.0 -webkit- |
| backface-visibility | 36.0  12.0 -webkit- | 10.0 | 16.010.0 -moz- | 23.015.0 -webkit- | 9.04.0 -webkit- |

The 3D rotateX() method (X-axis rotation)

The CSS 3D rotateX() method is used to rotate an element around its X-axis according to the given degree.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. width: 300px;
7. height: 100px;
8. background-color: lightpink;
9. border: 1px solid black;
10. }
11. div#myDiv {
12. -webkit-transform: rotateX(150deg); /\* Safari \*/
13. transform: rotateX(150deg); /\* Standard syntax \*/
14. }
15. **</style>**
16. **</head>**
17. **<body>**
18. **<div>**
19. This is JavaTpoint!
20. **</div>**
21. **<div** id="myDiv"**>**
22. This is JavaTpoint!
23. **</div>**
24. **<p><b>**Note:**</b>** Internet Explorer 9 (and earlier versions) does not support the rotateX() method.**</p>**
25. **</body>**
26. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-3d-transforms1)

The 3D rotateY() method (Y-axis rotation)

The CSS 3D rotateY() method is used to rotate an element around its Y-axis according to the given degree.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. width: 300px;
7. height: 100px;
8. background-color:lightpink;
9. border: 1px solid black;
10. }
12. div#myDiv {
13. -webkit-transform: rotateY(150deg); /\* Safari \*/
14. transform: rotateY(150deg); /\* Standard syntax \*/
15. }
16. **</style>**
17. **</head>**
18. **<body>**
19. **<div>**
20. Welcome to JavaTpoint!.
21. **</div>**
22. **<div** id="myDiv"**>**
23. Welcome to JavaTpoint!.
24. **</div>**
25. **<p><b>**Note:**</b>** Internet Explorer 9 (and earlier versions) does not support the rotateY() method.**</p>**
26. **</body>**
27. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-3d-transforms2)

The 3D rotateZ() method (Z-axis rotation)

The CSS 3D rotateZ() method is used to rotate an element around its Z-axis according to the given degree.

**See this example:**>

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. width: 300px;
7. height: 100px;
8. background-color: yellow;
9. border: 1px solid black;
10. }
11. div#myDiv {
12. -webkit-transform: rotateZ(90deg); /\* Safari \*/
13. transform: rotateZ(90deg); /\* Standard syntax \*/
14. }
15. **</style>**
16. **</head>**
17. **<body>**
18. **<div>**
19. Welcome to JavaTpoint!
20. **</div>**
21. **<div** id="myDiv"**>**
22. Welcome to JavaTpoint!
23. **</div>**
24. **<p><b>**Note:**</b>** Internet Explorer 9 (and earlier versions) does not support the rotateZ() method.**</p>**
25. **</body>**
26. **</html>**

CSS User Interface

CSS provides many user interface features like resizing elements, outlines and box sizing.

Following is a list of some common properties of CSS3 user interface:

|  |  |
| --- | --- |
| **Values** | **Description** |
| appearance | It facilitates users to make elements as user interface elements. |
| box-sizing | It facilitates users to fix elements on area in clear way. |
| icon | It is used to provide the icon on area. |
| resize | It is used to resize elements which are on area. |
| outline-offset | It is used to set space between an outline and the edge or border of an element. |
| nav-down | It is used to move down while pressing the down arrow button in keypad. |
| nav-left | It is used to move left while pressing the left arrow button in keypad. |
| nav-right | It is used to move right while pressing the right arrow button in keypad. |
| nav-up | It is used to move up while pressing the up arrow button in keypad. |

**Note:**resize and outline-offset are the most important properties of the CSS user interface. Resize property can have 3 common values:

* Horizontal resize
* Vertical resize
* Both (horizontal & vertical) resize.

CSS3 Resize property

The CSS3 resize property specifies that whether an element should be resized by the user or not.

Horizontal Resize

Let's take an example to resize the width of a <div> element. (Horizontal resize)

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. border: 2px solid;
7. padding: 20px;
8. width: 300px;
9. resize: horizontal;
10. overflow: auto;
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<p><b>**Note:**</b>** Internet Explorer does not support the resize property.**</p>**
16. **<div>**This example specifies how to resize the width of a div element.**</div>**
17. **</body>**
18. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-user-interface1)

**Example2:**

Vertical Resize

Let's take an example to resize the height of a <div> element. (Vertical resize)

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. border: 2px solid;
7. padding: 20px;
8. width: 300px;
9. resize: vertical;
10. overflow: auto;
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<p><b>**Note:**</b>** Internet Explorer does not support the resize property.**</p>**
16. **<div>**This example specifies how to resize the height of a div element.**</div>**
17. **</body>**
18. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-user-interface2)

Both (horizontal & vertical) resize

You can also resize the width and height of a <div> element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. border: 2px solid;
7. padding: 20px;
8. width: 300px;
9. resize: both;
10. overflow: auto;
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<p><b>**Note:**</b>** Internet Explorer does not support the resize property.**</p>**
16. **<div>**This example specifies how to resize both the height and the width of this div element.**</div>**
17. **</body>**
18. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-user-interface3)

CSS3 Outline Offset

The outline-offset property is used to add space between an outline and border of an element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. margin: 20px;
7. padding: 10px;
8. width: 300px;
9. height: 100px;
10. border: 1px solid black;
11. outline: 1px solid red;
12. outline-offset: 10px;
13. }
14. **</style>**
15. **</head>**
16. **<body>**
17. **<p><b>**Note:**</b>** Internet Explorer does not support the outline-offset property.**</p>**
18. **<div>**This example specifies an outline 10px outside the border edge.**</div>**
19. **</body>**
20. **</html>**

CSS Layout

**CSS layout** is easy to design. We can use CSS layout to design our web page such as home page, contact us, about us etc.

There are 3 ways to design layout of a web page:

1. **HTML Div with CSS**: fast and widely used now.
2. **HTML Table**: slow and less preferred.
3. **HTML Frameset**: deprecated now.

A CSS layout can have header, footer, left pane, right pane and body part. Let's see a simple example of CSS layout.

CSS layout example

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. .header{margin:-8px -8px 0px;background-image:linear-gradient(145deg,#7379ff,#b524ef);color:white;text-align:center;padding:10px;}
6. .container{width:100%}
7. .left{width:15%;float:left;}
8. .body{width:65%;float:left;background-color:pink;padding:5px;}
9. .right{width:15%;float:left;}
10. .footer{margin:-8px;clear:both;background-image:linear-gradient(145deg,#7379ff,#b524ef);color:white;text-align:center;padding:10px;}
11. **</style>**
12. **</head>**
13. **<body>**
14. **<div** class="header"**><h2>**JavaTpoint**</h2></div>**
16. **<div** class="container"**>**
17. **<div** class="left"**>**
18. **<p>**Left Page**</p>**
19. **</div>**
20. **<div** class="body"**>**
21. **<h1>**Body Page**</h1>**
22. **<p>**Page Content goes here**</p><p>**Page Content goes here**</p><p>**Page Content goes here**</p>**
23. **<p>**Page Content goes here**</p><p>**Page Content goes here**</p><p>**Page Content goes here**</p>**
24. **<p>**Page Content goes here**</p><p>**Page Content goes here**</p><p>**Page Content goes here**</p>**
25. **<p>**Page Content goes here**</p><p>**Page Content goes here**</p><p>**Page Content goes here**</p>**
26. **<p>**Page Content goes here**</p>**
27. **</div>**
28. **<div** class="right"**>**
29. **<p>**Right Page**</p>**
30. **</div>**
31. **</div>**
33. **<div** class="footer"**>**
34. **<p>**Footer**</p>**
35. **</div>**
37. **</body>**
38. **</html>**

CSS Table

We can apply style on HTML tables for better look and feel. There are some CSS properties that are widely used in designing table using CSS:

* border
* border-collapse
* padding
* width
* height
* text-align
* color
* background-color

CSS Table Border

We can set border for the table, th and td tags using the CSS border property.

1. **<style>**
2. table, th, td {
3. border: 1px solid black;
4. }
5. **</style>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=htmltable4)

Output:

|  |  |  |
| --- | --- | --- |
| **First\_Name** | **Last\_Name** | **Marks** |
| Sonoo | Jaiswal | 60 |
| James | William | 80 |
| Swati | Sironi | 82 |
| Chetna | Singh | 72 |

CSS Table Border Collapse

By the help of border-collapse property, we can collapse all borders in one border only.

1. **<style>**
2. table, th, td {
3. border: 2px solid black;
4. border-collapse: collapse;
5. }
6. **</style>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=htmltable5)

Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Last Name** | **Marks** |
| Sonoo | Jaiswal | 60 |
| James | William | 80 |
| Swati | Sironi | 82 |
| Chetna | Singh | 72 |

CSS Table Padding

We can specify padding for table header and table data using the CSS padding property.

1. **<style>**
2. table, th, td {
3. border: 1px solid black;
4. border-collapse: collapse;
5. }
6. th, td {
7. padding: 10px;
8. }
9. **</style>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=htmltable6)

Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Last Name** | **Marks** |
| Sonoo | Jaiswal | 60 |
| James | William | 80 |
| Swati | Sironi | 82 |
| Chetna | Singh | 72 |

CSS Table: Styling even and odd cells

We can style even and odd table cells for better look and feel. In this code, we are displaying different background colors on even and odd cells. Moreover, we have changed the background-color and color of <th> tag.

CSS code:

1. **<style>**
2. table, th, td {
3. border: 1px solid black;
4. border-collapse: collapse;
5. }
6. th, td {
7. padding: 10px;
8. }
9. table#alter tr:nth-child(even) {
10. background-color: #eee;
11. }
12. table#alter tr:nth-child(odd) {
13. background-color: #fff;
14. }
15. table#alter th {
16. color: white;
17. background-color: gray;
18. }
19. **</style>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=htmltable8)

Output:

