HTML 5 and CSS3

https://chools.in/   https://choolsgroup.com/   https://choolskool.com/

Content

- What is HTML5
- Components of HTML 5
- Detecting HTML 5 capabilities
- HTML5 Form tags
- Media tags
- Web worker
- Canvas
- Geolocation
- Web Storage
- Introduction to CSS3
- Introduction to Media element
- CSS3 validations
- Many more….

What is HTML 5

- HTML5 will be the new standard for HTML.

- The previous version of HTML, HTML 4.01, came in 1999. The web has changed a lot since then.

- HTML5 is still a work in progress. However, the major browsers support many of the new HTML5 elements and APIs.

- HTML5 introduces a number of new elements and attributes that helps in building a modern websites

- HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

What is New in HTML 5

- **New Semantic Elements:** These are like <header>, <footer>, and <section>.

- **Forms 2.0:** Improvements to HTML web forms where new attributes have been introduced for <input> tag.

- **Persistent Local Storage:** To achieve without resorting to third-party plugins.

What is New in HTML 5

- **Canvas:** This supports a two-dimensional drawing surface that you can program with JavaScript.

- **Audio & Video:** You can embed audio or video on your web pages without resorting to third-party plugins.

- **Geolocation:** Now visitors can choose to share their physical location with your web application.

HTML 5 Syntax

- **The DOCTYPE:**
- DOCTYPEs in older versions of HTML were longer because the HTML language was SGML based and therefore required a reference to a DTD.

- HTML 5 authors would use simple syntax to specify DOCTYPE as follows:

- <!DOCTYPE html> All the above syntax is case-insensitive.

- **Character Encoding:**
- HTML 5 authors can use simple syntax to specify Character Encoding as follows:
- <meta charset="UTF-8"> All the above syntax is case-insensitive.

HTML 5 Syntax

- **The <script> tag:**
- It's common practice to add a type attribute with a value of "text/javascript" to script elements as follows:

- <script type="text/javascript" src="scriptfile.js"></script> HTML 5 removes extra information required and you can use simply following syntax:
- <script src="scriptfile.js"></script>

- **The <link> tag:**
- So far you were writing <link> as follows:
- <link rel="stylesheet" type="text/css" href="stylefile.css"> HTML 5 removes extra information required and you can use simply following syntax:
- <link rel="stylesheet" href="stylefile.css">

HTML 5 Elements

New types for <input> tag:

The input element's type attribute now has the following new values:

Type	Description
color	Color selector, which could be represented by a wheel or swatch picker
date	Selector for calendar date
datetime-local	Date and time display, with no setting or indication for time zones
datetime	Full date and time display, including a time zone.
email	Input type should be an email.
month	Selector for a month within a given year
number	A field containing a numeric value only
range	Numeric selector within a range of values, typically visualized as a slider
search	Term to supply to a search engine. For example, the search bar atop a browser.
tel	Input type should be telephone number.
time	Time indicator and selector, with no time zone information
url	Input type should be URL type.
week	Selector for a week within a given year

HTML 5 Removed Elements

Removed Elements

The following HTML 4.01 elements are removed from HTML5:

<acronym>
<applet>
<basefont>
<big>
<center>
<dir>
<font>
<frame>
<frameset>
<noframes>
<strike>
<tt>

Tags - <header>:

"a group of introductory or navigational aids. A header element typically contains the section’s heading (an h1–h6 element or an hgroup element), but can also contain other content, such as a table of contents, a search form, or any relevant logos."

```
<header>
    <h1>Main Header</h1>
    <p class="tagline">Tagline</p>
</header>
```

Tags - <nav>:

"Not all groups of links on a page need to be in a nav element only sections that consist of major navigation blocks are appropriate for the nav element."

```
<nav role="navigation">
  <ul>
    <li>
      <a href="#" title="link">link</a>
    </li>
  </ul>
</nav>
```

Tags - <article>:

"a composition that forms an independent part of a document, page, application, or site. This could be a forum post, a magazine or newspaper article, a Web log entry ..."

```
<article>
    <h2>Item</h2>
    <p>Some content here.</p>
</article>
```

Tags - <section>:

"section is a blob of content that you could store as an individual record in a database."

```
<section id="foo">
   <h2>Foo</h2>
   <p>
       Content here
   </p>
</section>
```

Tags - <article> vs. <section>:

"Authors are encouraged to use the article element instead of the section element when it would make sense to syndicate the contents of the element."

"Section is used when there is naturally a heading at the start of the section."

Tags - <footer>:

"The footer element represents a footer for its nearest ancestor sectioning content or sectioning root element. A footer typically contains information about its section such as who wrote it, links to related documents, copyright data, and the like."

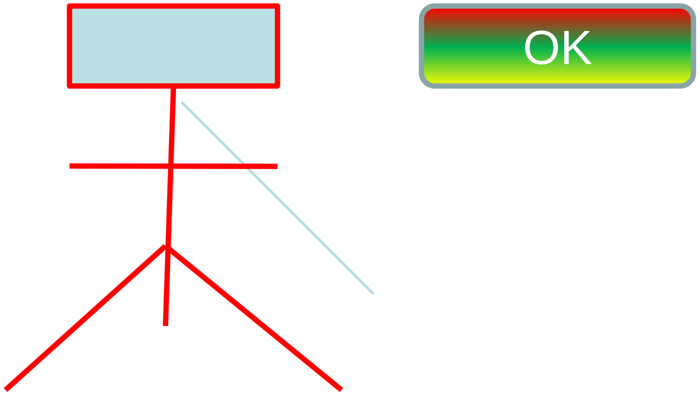
Tags - <footer>:

```
<footer role="contentinfo">
    <p>
        Footer
    </p>
    <nav>
        <h5>Quick Links</h5>
        <ul role="navigation">
            <li>
                <a href="#">link</a>
            </li>
        </ul>
    </nav>
</footer>
```

Canvas

HTML5 element <canvas> gives you an easy and powerful way to draw graphics using JavaScript.
It can be used to draw graphs, make photo compositions or do simple (and not so simple) animations.

```
<!DOCTYPE HTML>
<html>
<head>
<style>
#mycanvas{
   border:1px solid red;
}
</style>
</head>
<body>
   <canvas id="mycanvas" width="100" height="100"></canvas>
</body>
</html>
```



Video

- **HTML5 <video>**
- HTML5 has DOM methods, properties, and events for the <video> and <audio> elements.

- There are methods for playing, pausing, and loading, for example and there are properties (like duration and volume).
- There are also DOM events that can notify you when the <video> element begins to play, is paused, is ended, etc.

Audio

```
<!DOCTYPE html>
<html>
<body>

<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
Your browser does not support the audio element.
</audio>

</body>
</html>
```

Geolocation

HTML5 Geolocation API lets you share your location with your favorite web sites.

A Javascript can capture your latitude and longitude and can be sent to backend web server and do fancy location-aware things like finding local businesses or showing your location on a map.

Today most of the browsers and mobile devices support Geolocation API. The geolocation APIs work with a new property of the global navigator object ie. Geolocation object which can be created as follows:

```
var geolocation = navigator.geolocation;
```

The geolocation object is a service object that allows widgets to retrieve information about the geographic location of the device.

Geolocation

```
<script>
var x=document.getElementById("demo");
function getLocation()
  {
  if (navigator.geolocation)
    {
    navigator.geolocation.getCurrentPosition(showPosition);
    }
  else{x.innerHTML="Geolocation is not supported by this
browser.";}
  }
function showPosition(position)
  {
  x.innerHTML="Latitude: " + position.coords.latitude +
  "<br>Longitude: " + position.coords.longitude;
  }
</script>
```

Web Workers

A web worker is a JavaScript running in the background, without affecting the performance of the page.

What is a Web Worker?

When executing scripts in an HTML page, the page becomes unresponsive until the script is finished.

A web worker is a JavaScript that runs in the background, independently of other scripts, without affecting the performance of the page.

You can continue to do whatever you want: clicking, selecting things, etc., while the web worker runs in the background.

Web Workers

```
function startWorker()
{
if(typeof(Worker)!=="undefined")
  {
  if(typeof(w)=="undefined")
  {
  w=new Worker("demo_workers.js");
  }
  w.onmessage = function (event) {
    document.getElementById("result1").innerHTML=event.data;
    };
  }else  {
  document.getElementById("result").innerHTML="Sorry, your browser does
not support Web Workers...";
  }
}
function stopWorker()
{
w.terminate();
}
```

Web Storage

With HTML5, web pages can store data locally within the user's browser.

Earlier, this was done with cookies. However, Web Storage is more secure and faster.
The data is not included with every server request, but used ONLY when asked for. It is also possible to store large amounts of data, without affecting the website's performance.

The data is stored in key/value pairs, and a web page can only access data stored by itself.

Web Storage

localStorage and sessionStorage

There are two new objects for storing data on the client:

localStorage - stores data with no expiration date

sessionStorage - stores data for one session

Before using web storage, check browser support for localStorage and sessionStorage:

```
if(typeof(Storage)!=="undefined")
  {
  // Yes! localStorage and sessionStorage support!
  // Some code.....
  }
else
  {
  // Sorry! No web storage support..
  }
```

Web Storage

localStorage and sessionStorage
There are two new objects for storing data on the client:

localStorage - stores data with no expiration date

sessionStorage - stores data for one session

Before using web storage, check browser support for localStorage and sessionStorage:

```
if(typeof(Storage)!=="undefined")
  {
  // Yes! localStorage and sessionStorage support!
  // Some code.....
  }
else
  {
  // Sorry! No web storage support..
  }
```

CSS3

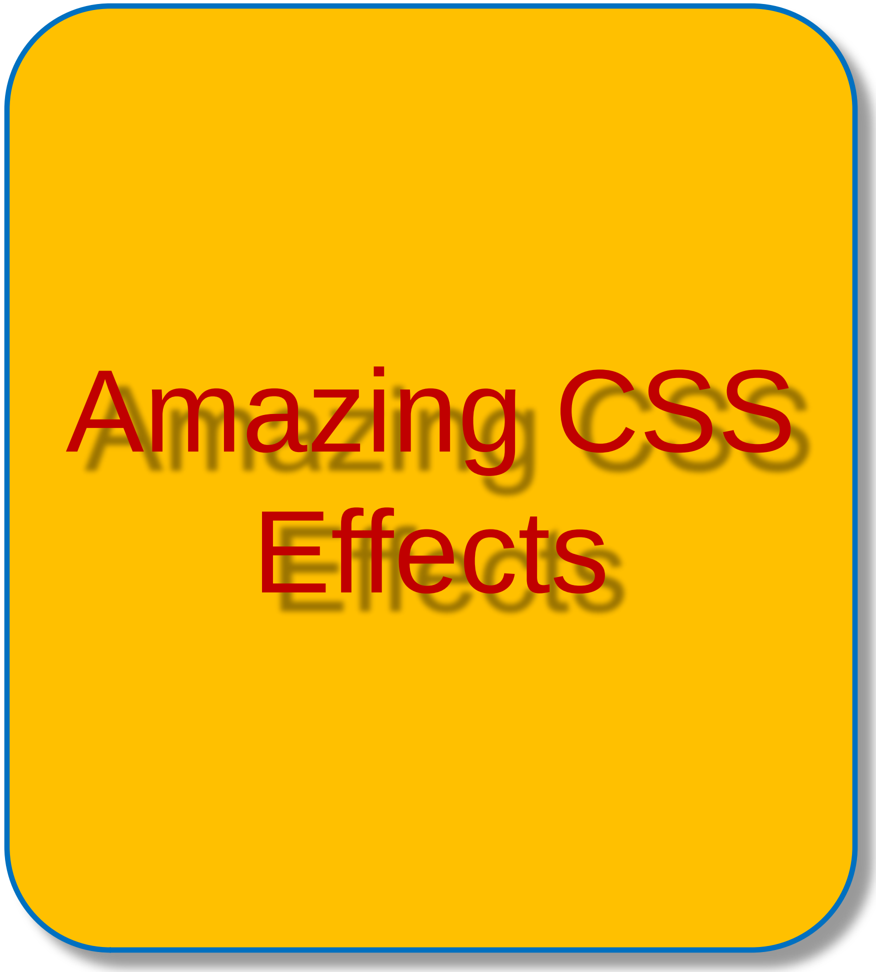
CSS3 is split up into "modules". The old specification has been split into smaller pieces, and new ones are also added.

- Some of the most important CSS3 modules are:
- Selectors
- Box Model
- Backgrounds and Borders
- Text Effects
- 2D/3D Transformations
- Animations
- Multiple Column Layout
- User Interface

CSS Effects

- Rounded corners
- Gradients
- Box and text shadows
- Fonts
- Transparencencies
- Multiple background images and border images
- Multiple columns and grid layout
- Box sizing
- Stroke and outlines
- Animation, movement and rotation
- Improved selectors

CSS Effect Example



```
.amazing {
border: 1px solid blue;
color: red;
background-color: gold;

-webkit-border-radius: 40px;
-moz-border-radius: 40px;
border-radius: 40px;

-webkit-box-shadow: 8px 8px 6px
    #474747;
-moz-box-shadow: 8px 8px 6px
    #474747;
box-shadow: 8px 8px 6px #474747;

text-shadow: 8px 8px 2px #595959;
filter: dropshadow(color=#595959,
    offx=8, offy=8);
}
```

Create Rounded Corners

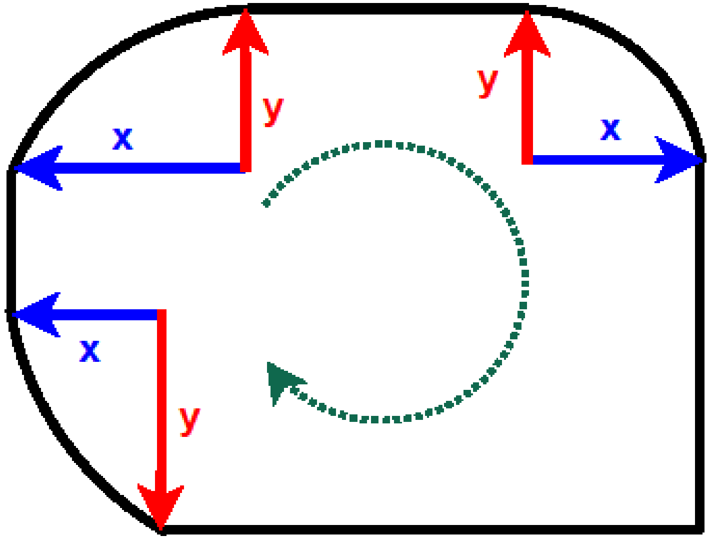
border -*-*-radius: [**x**] [**y**]?

x = **horizontal** radius [<*length*> | <*%*>]

y = **vertical** radius [<*length*> | <*%*>]

border-top-left radius: x y;

border-top-right-radius: x y;



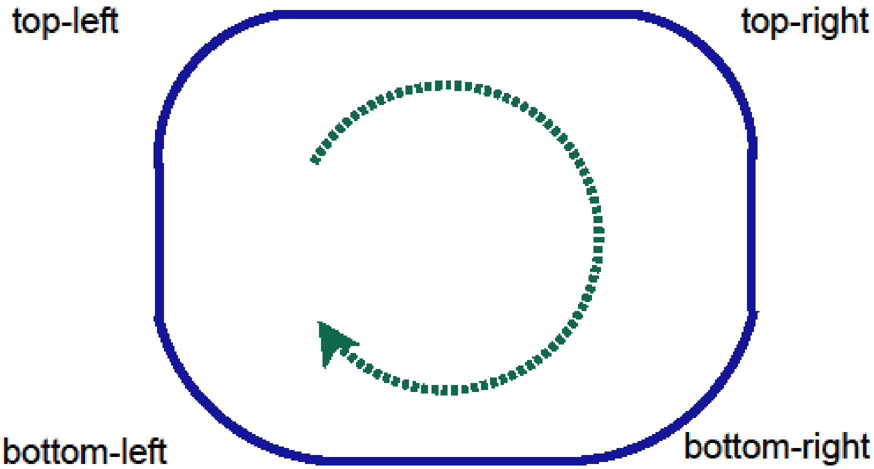
border-bottom-left-radius: x y;

border-bottom-right-radius: 0;

Create Rounded Corners

border-radius: [x] {1,4} [/ [y] {1,4}]?

border-radius: 5px 10px 10px 10px / 10px 10px 5px 5px;
border-radius: 5px 10px / 10px;
border radius: 10px;



Create Rounded Corners

W3C Specification	Mozilla Implementation
border-radius	-moz-border-radius
border-top-left-radius	-moz-border-radius-topleft
border-top-right-radius	-moz-border-radius-topright
border-bottom-right-radius	-moz-border-radius-bottomright
border-bottom-left-radius	-moz-border-radius-bottomleft

Drop Shadow

text-shadow: **x** **y** b **color**

- **x** = horizontal offset
  - **x** < 0: left of the text
  - **x** > 0: right of the text
- **y** = vertical offset
  - **y** < 0: above the text
  - **y** > 0: below the text
- **b** = blur radius

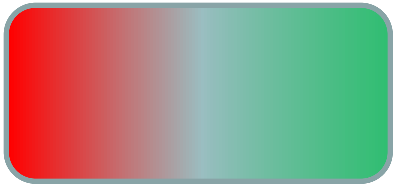
Drop Shadow

box-shadow: **x** **y** **b** **color**

Examples**:**

```
-webkit-box-shadow: -3px 3px 3px #999999;
-moz-box-shadow: -3px 3px 3px #999999;
text-shadow: -4px 4px 3px #999999;
```

Create Gradient Buttons



- rgba(r, g, b, opacity)
- background:-webkit-gradient(linear,0% 0%,0% 100%, from(rgba(255,255,255,1)),to(rgba(185,185,185,1)));
- background:-moz-linear-gradient(top,rgba(255,255,255,1), rgba(185,185,185,1));

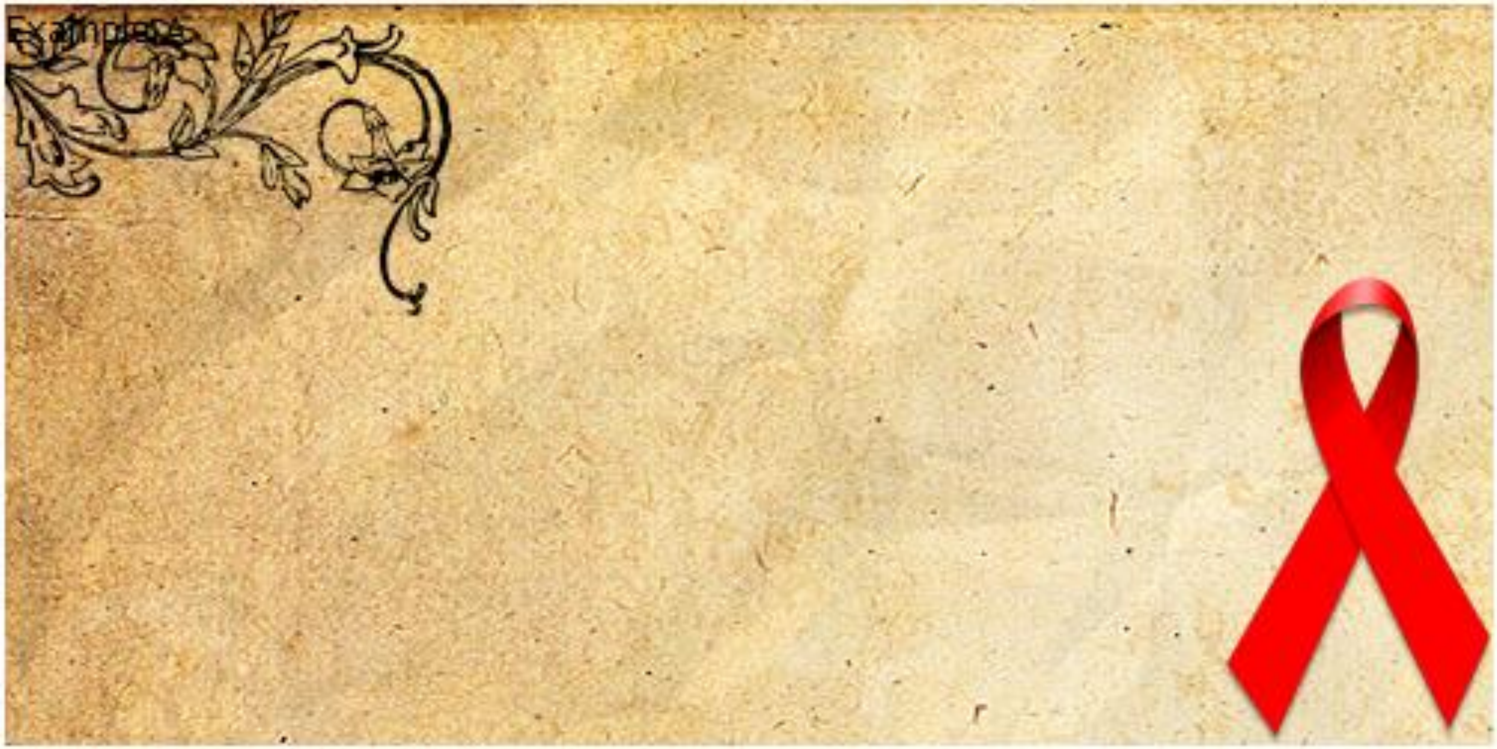
Multiple background images

```
background:
<image> <position> <size> <repeat> <attachment> <box>,
<image> <position> <size> <repeat> <attachment> <box>,
<image> <position> <size> <repeat> <attachment> <box>,
<image> <position> <size> <repeat> <attachment> <box> <color>;
```

OR

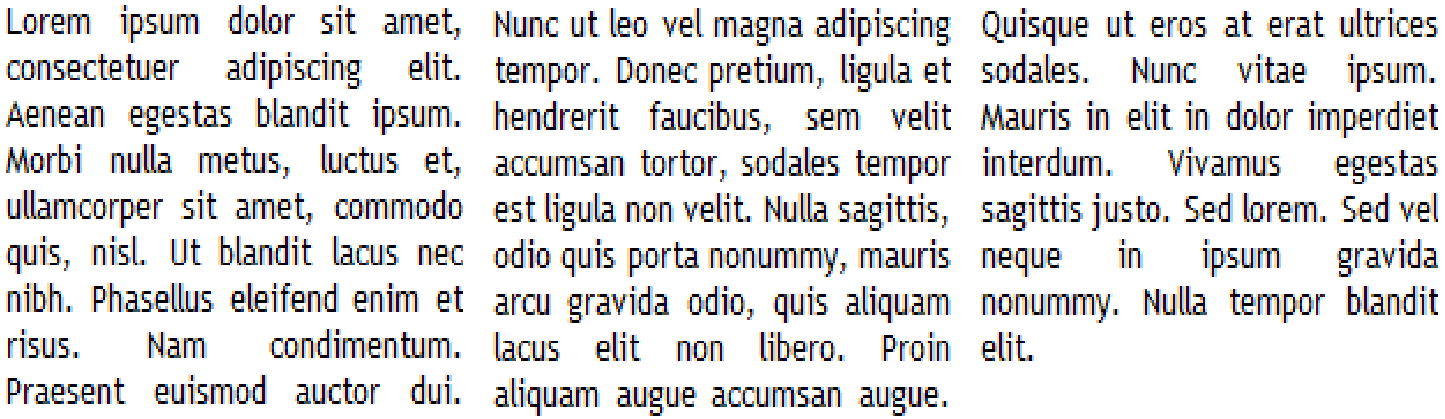
```
background-image: <image>, <image>, <image>, <image>;
background-repeat: <repeat>, <repeat>, <repeat>,<repeat>;
background-position: <position>, <position>, <position>,<position>;
/* plus any background attachment and/or box properties as needed */
```

```
#example {
    width: 500px;
    height: 250px;
    background-image: url(decoration.png), url(ribbon.png),
    url(old_paper.jpg);
    background-repeat: no-repeat;
    background-position: left top, right bottom, left top;
    }
```



Multi-Column Layout

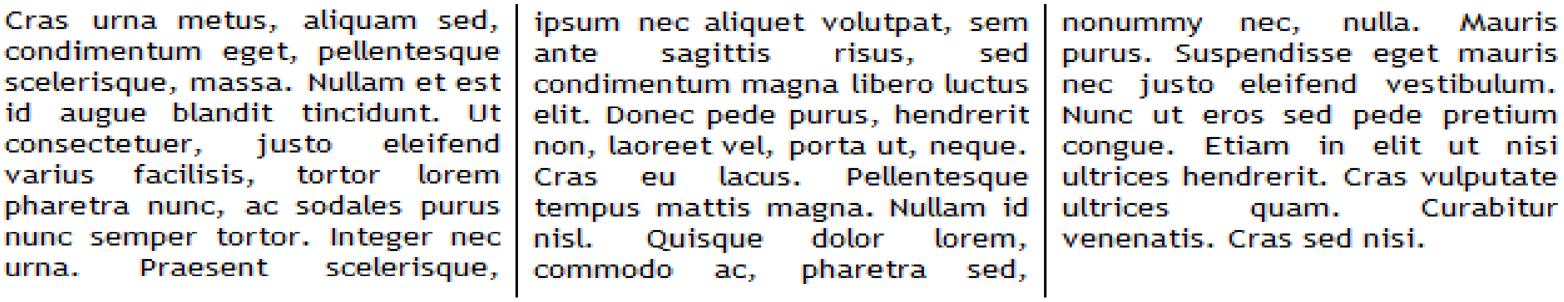
```
#multi-column {
/* For Mozilla: */
    -moz-column-width: 13em;
    -moz-column-gap: 1em;
 /* For WebKit: */
    -webkit-column-width: 13em;
    -webkit-column-gap: 1em;
}
```



Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean egestas blandit ipsum. Morbi nulla metus, luctus et, ullamcorper sit amet, commodo quis, nisl. Ut blandit lacus nec nibh. Phasellus eleifend enim et risus. Nam condimentum. Praesent euismod auctor dui. Nunc ut leo vel magna adipiscing tempor. Donec pretium, ligula et hendrerit faucibus, sem velit accumsan tortor, sodales tempor est ligula non velit. Nulla sagittis, odio quis porta nonummy, mauris arcu gravida odio, quis aliquam lacus elit non libero. Proin aliquam augue accumsan augue. Quisque ut eros at erat ultrices sodales. Nunc vitae ipsum. Mauris in elit in dolor imperdiet interdum. Vivamus egestas sagittis justo. Sed lorem. Sed vel neque in ipsum gravida nonummy. Nulla tempor blandit elit.

Multi-Column Layout

```
#multi-column {
      -moz-column-count: 3;
     -moz-column-gap: 1em;
     -moz-column-rule: 1px solid black;
     -webkit-column-count: 3;
     -webkit-column-gap: 1em;
     -webkit-column-rule: 1px solid black;
}
```



Cras urna metus, aliquam sed, condimentum eget, pellentesque scelerisque, massa. Nullam et est id augue blandit tincidunt. Ut consectetuer, justo eleifend varius facilisis, tortor lorem pharetra nunc, ac sodales purus nunc semper tortor. Integer nec urna. Praesent scelerisque, ipsum nec aliquet volutpat, sem ante sagittis risus, sed condimentum magna libero luctus elit. Donec pede purus, hendrerit non, laoreet vel, porta ut, neque. Cras eu lacus. Pellentesque tempus mattis magna. Nullam id nisl. Quisque dolor lorem, commodo ac, pharetra sed, nonummy nec, nulla. Mauris purus. Suspendisse eget mauris nec justo eleifend vestibulum. Nunc ut eros sed pede pretium congue. Etiam in elit ut nisi ultrices hendrerit. Cras vulputate ultrices quam. Curabitur venenatis. Cras sed nisi.

Demos : CSS3