

CSS Text Effect:

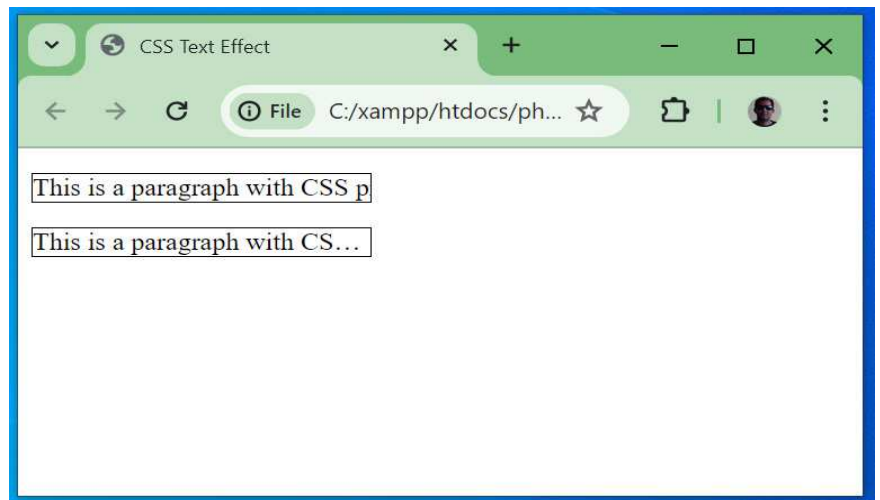
We can apply different effects on the text used within an HTML document. These CSS text-effect properties are very useful and also make the text stylish and more attractive.

- text-overflow
- word-break
- word-wrap
- writing-mode

Example of text-overflow:

```
<html>
<head>
<title>CSS Text Effect</title>
<style>
p.p1 {
  white-space: nowrap;
  width: 200px;
  border: 1px solid #000000;
  overflow: hidden;
  text-overflow: clip;
}
p.p2 {
  white-space: nowrap;
  width: 200px;
  border: 1px solid #000000;
  overflow: hidden;
  text-overflow: ellipsis;
}
</style>
</head>
<body>
<p class="p1">This is a paragraph with CSS property text-overflow and value clip.</p>
<p class="p2">This is a paragraph with CSS property text-overflow and value ellipsis.</p>
</body>
</html>
```

Output:



writing-mode:

The CSS writing-mode property specifies whether the text should be placed **horizontally** or **vertically**. This property comes with three different values **horizontal-tb**, **vertical-rl**, and **vertical-lr**.

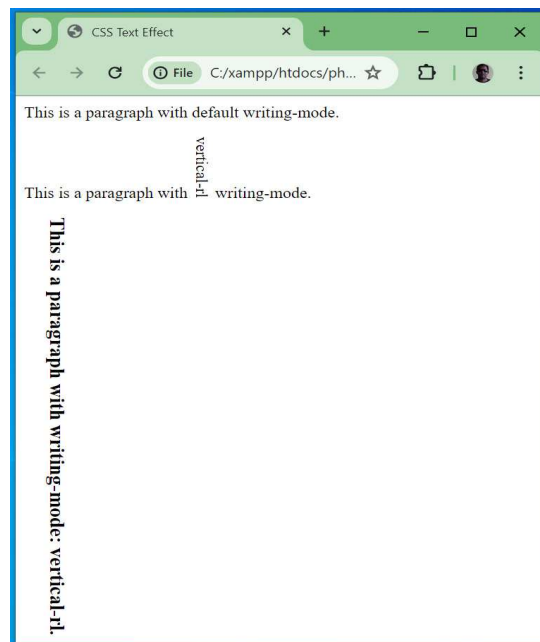
Example of writing-mode:

```
<html>
<head>
<title>CSS Text Effect</title>
<style>
p.p1 {
    writing-mode: horizontal-tb;
}

span.p2 {
    writing-mode: vertical-lr;
}

p.p2 {
    writing-mode: vertical-rl;
    font-size: 20px;
    font-weight: bold;
}
</style>
</head>
<body>
<p class="p1">This is a paragraph with default writing-mode.</p>
<p>This is a paragraph with <span class="p2">vertical-rl</span>writing-mode.</p>
<p class="p2">This is a paragraph with writing-mode: vertical-rl.</p>
</body>
</html>
```

Output:



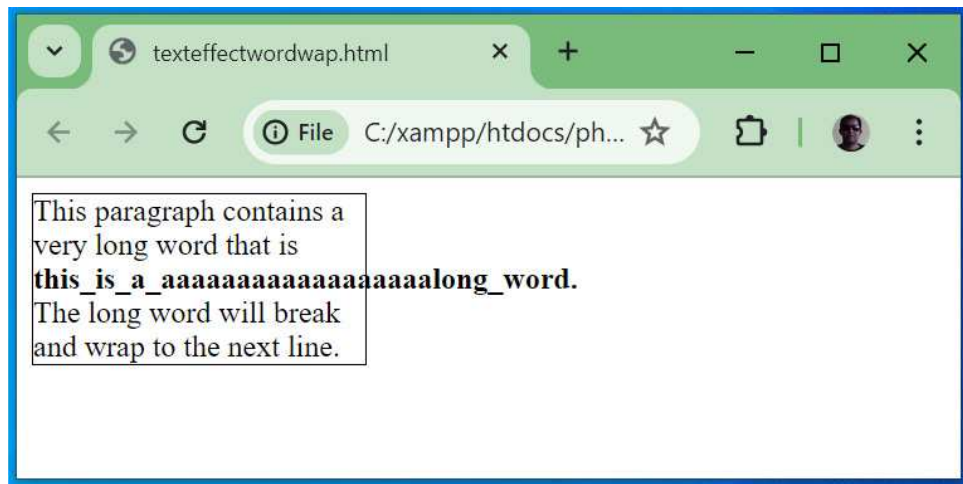
word-wrap:

CSS word-wrap property allows us to wrap long words into the next line **without breaking** them into **two or more words**. This property avoids word overflow when a string is too long to fit in a container box. This property comes with multiple values, these are normal, break-word, initial, and inherit.

Example of word-wrap:

```
<html>
<head>
<style>
p.pl {
    width: 11em;
    border: 1px solid #000000;
    word-wrap: initial;
}
</style>
</head>
<body>
    <p class="pl">This paragraph contains a very long word that is <b>
this_is_a_aaaaaaaaaaaaaaaaaalong_word.</b>The long word will break and wrap to the next line.
    </p>
</body>
</html>
```

Output:



word-break:

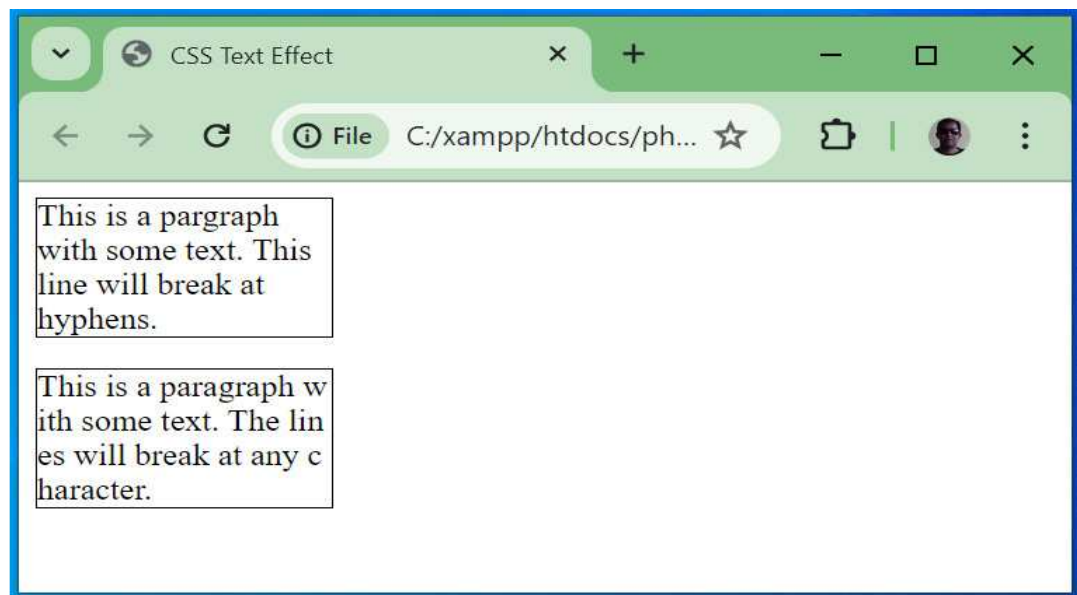
The CSS word-break property determines the rules for how words should break at the end of the line. The default value of word-break is normal which is used automatically when the user has not assigned any value for it.

Example of word-break:

```
<html>
<head>
<title>CSS Text Effect</title>
<style>
p.p1 {
    width: 140px;
    border: 1px solid #000000;
    word-break: keep-all;
    word-break: initial;
}

p.p2 {
    width: 140px;
    border: 1px solid #000000;
    word-break: break-all;
}
</style>
</head>
<body>
    <p class="p1">This is a paragraph with some text. This line will
        break at hyphens.</p>
    <p class="p2">This is a paragraph with some text. The lines will
        break at any character.</p>
</body>
</html>
```

Output:



Responsive Web Design:

The responsive web design is a way to make a web page render well on all types of screen sizes and devices such as laptops, desktops, smartphones, tablets, etc. Any website which is created using the **responsive web design (RWD)** can automatically adjust the **layout** and the **appearance** according to the environment in which it is viewed.



Viewport: The user's visible area of a webpage is called viewport. The viewport varies with the device and will be smaller on a mobile phones, web pages were designed only for computer screens and it was common for web pages to have a static design and a fixed size. But now we need web pages that changes according to the viewport of device to look good.

Setting the Viewport

Pages optimized for a variety of devices must include a meta viewport tag in the document's head. Add the following <meta> tag to all your web pages:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Responsive Text Size: You can make the text responsive by using the viewport-width property.

Example:

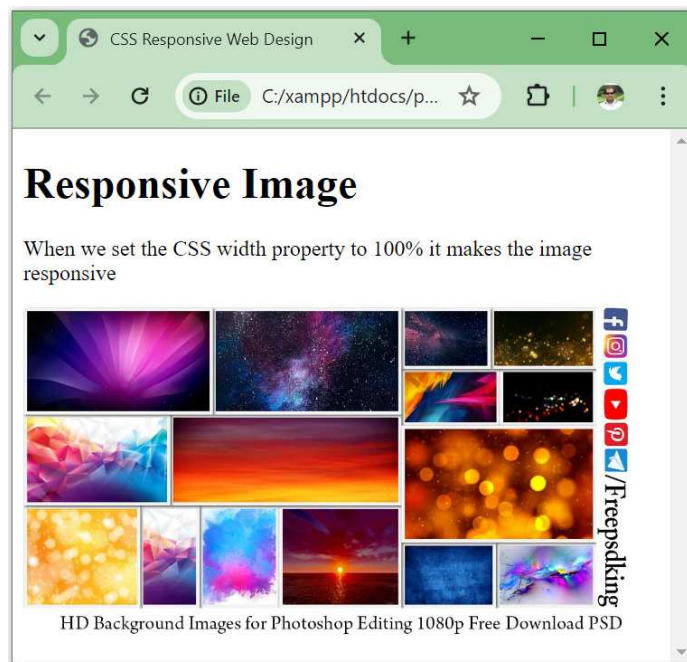
```
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>CSS Responsive Web Design</title>
<style>
.heading {
    font-size: 20;
    color: #f30505;
}
#paragraph {
    font-size: 50px;
}
</style>
</head>
<body>
<h1 class="heading">This is headings</h1>
<P id="paragraph">Resize the browser window to see how the text
    size change.</P>
</body></html>
```

Responsive Images: Set the width to 100% to make the image responsive and scale up and down.

Example:

```
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>CSS Responsive Web Design</title>
</head>
<body>
  <h1>Responsive Image</h1>
  <p>When we set the CSS width property to 100% it makes the image
    responsive</p>
  
</body>
</html>
```

Output:



Media queries:

Media queries are used to provide different styles for different size devices such as mobile, desktop etc.

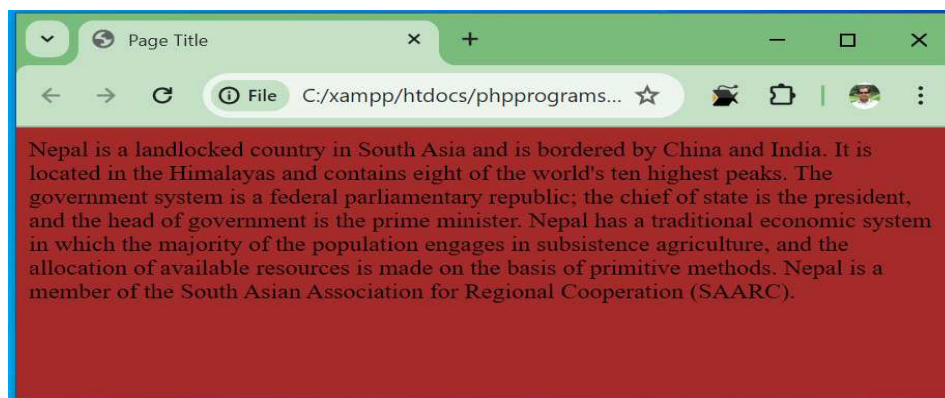
Example:

```
<html>
<head>

<title>Page Title</title>
<meta name='viewport' content='width=device-width, initial-scale=1'>
<style>
body {
    background-color: brown;
}

@media only screen and (max-width: 600px) {
    body {
        background-color: aquamarine;
    }
}
</style>
</head>
<body>
    <p>Nepal is a landlocked country in South Asia and is bordered by
        China and India. It is located in the Himalayas and contains eight of
        the world's ten highest peaks. The government system is a federal
        parliamentary republic; the chief of state is the president, and the
        head of government is the prime minister. Nepal has a traditional
        economic system in which the majority of the population engages in
        subsistence agriculture, and the allocation of available resources is
        made on the basis of primitive methods. Nepal is a member of the South
        Asian Association for Regional Cooperation (SAARC).</p>
</body>
</html>
```

Output:



Introduction to Bootstrap:

Bootstrap is an open source JavaScript framework developed by Twitter's team of designers and developers. It is the most popular HTML, CSS, and JS framework for developing responsive, mobile friendly website. It is also called as a front-end-framework. It contains HTML, CSS-based design templates for typography, forms, buttons, and navigation and other interface components, as well as optional JavaScript extensions.

Why use Bootstrap?

- It is very easy to use (If you have the basic knowledge of HTML and CSS you can use Bootstrap)
- The user can develop responsive websites.
- Supported browsers are Google Chrome, Mozilla Firefox, Internet Explorer, Safari and Opera etc.
- It is easy to getting started.
- To create responsive grid system.
- Bundled JavaScript plug-ins.
- List of components to use (Typography, Code, Table, Forms, Button, Images and Icons etc.)
- Good documentation.
- Responsive design.
- Mobile first approach.
- Open source.
- Saves time, customizable.

Installation:

Bootstrap can be installed in our web page in two ways, one by downloading compiled code for bootstrap which includes compiled and minified CSS bundles as well as compiled and minified JavaScript plugins but does not include documentation, source files or any optional JavaScript dependencies (jQuery and popper.js).

Another method by which we can install bootstrap on our web page is by using bootstrap CDN. Bootstrap CDN is a link that connects web page with bootstrap. We can get this CDN link from getbootstrap.com and copy and paste that link in the <head> tag of our web page. Using CDN link is better in case of performance. Once we have used CDN link for bootstrap in our web page now we can easily use bootstrap in our web page.