Unit - IV

4.1 CSS Formatting

Fonts

CSS ஃபான்ட் பண்புகள் ஒரு வலைப்பக்கத்தில் உள்ள உரையின் ஃபான்ட்களை அமைக்கப் பயன்படுகின்றன. சில முக்கியமான ஃபான்ட் பண்புகள்:

- font-family
- font-style
- font-weight
- font-size
- Google Fonts

font-family

இந்த பண்பு ஒரு உறுப்புக்கான ஃபான்ட்களின் பட்டியலை குறிப்பிடப் பயன்படுகிறது.

Syntax

font-family: <family-name>, <generic-name>;

- **family-name**: ஒரு குறிப்பிட்ட எழுத்துருவின் பெயர் மேற்கோள்களில் கொடுக்கப்பட்டுள்ளது
- **generic-name**: எழுத்துருவைக் குறிப்பிடுவதற்கு ஒரு பின்னடைவு பொறிமுறையாகப் பயன்படுத்தப்படுகிறது. பட்டியலில் கடைசி எழுத்துருவாக கொடுக்கப்பட்டுள்ளது.

Generic Font Names

- serif: **stroke** at end
- sans-serif: **no stroke** at end
- cursive: appears like **handwritten** text
- monospace: each character same `width`
- fantasy: **decorative** text

font-style

இந்த பண்பு எழுத்துரு பாணியை குறிப்பிடப் பயன்படுகிறது, எடுத்துக்காட்டாக italic (சாய்வு), oblique (சாய்வு), அல்லது normal (நேரான).

Syntax

font-style: italics;

font-weight

இந்த பண்பு எழுத்துருவின் தடிமனை குறிப்பிடப் பயன்படுகிறது, எடுத்துக்காட்டாக bold (தடித்த), bolder (அதிக தடித்த), lighter (மெலிந்த), normal (சாதாரண) அல்லது 100-900 வரம்பில் உள்ள எண்சார் மதிப்பு.

Syntax

font-weight: bold;

font-size

இந்த பண்பு எழுத்துருவின் அளவை px, em அல்லது % இல் அமைக்கப் பயன்படுகிறது. எழுத்துரு அளவு vw மதிப்பைப் பயன்படுத்தி viewport இன் அளவைப் பொறுத்து அமைக்கப்படலாம்.

Syntax

font-size: 5px;

Google Fonts

CSS, குறிப்பிட்ட இணைப்பு டேக்குகளைப் பயன்படுத்தி Google வழங்கும் ஃபான்ட்களைப் பயன்படுத்த அனுமதிக்கிறது. இது டெவலப்பர்கள் அவர்களின் systemல் இல்லாத ஃபான்ட்களை காண்பிக்க உதவுகிறது.

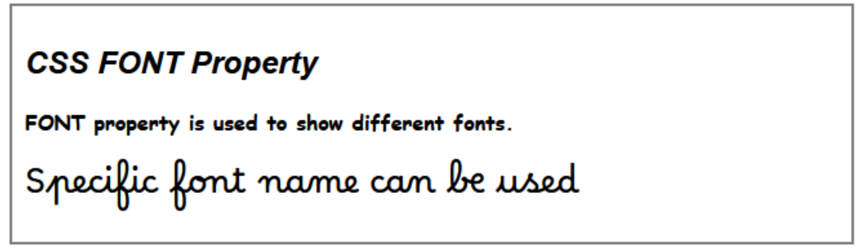
Syntax

```
<link rel="stylesheet"
  href="https://fonts.googleapis.com/css?family = Borel">
```

Example

```
h1{
    font-family:serif;
}
h2{
    font-family: sans-serif;
    font-style: italic;
}
p{
    font-family: cursive;
    font-weight: bold;
}
section{
    font-family: Borel;
    font-size: 4vw;
}
<article>
    <h2>CSS FONT Property</h2>
    <p>
        FONT property is used to show different fonts.
    </p>
    <section>
        Specific font name can be used
    </section>
</article>
```

Output



Kerning

Kerning என்பது ஒரு சொல்லில் உள்ள தனிப்பட்ட எழுத்துகளுக்கு இடையே உள்ள இடைவெளியை சரிசெய்து, படிக்கத்தக்க தன்மையையும் அழகியலையும் மேம்படுத்தும் செயல்முறையாகும். இது letter-spacing பண்பு மூலம் செயல்படுத்தப்படுகிறது.

Syntax

```
letter-spacing: spaceValue;
```

Leading

Leading in CSS, line height என்றும் அழைக்கப்படுகிறது, உரையின் வரிகளுக்கு இடையே உள்ள செங்குத்து இடைவெளியை குறிக்கிறது. இது line-height பண்பு மூலம் செயல்படுத்தப்படுகிறது.

Syntax

```
line-height: spaceValue;
```

Indenting

Indenting in CSS, உரையின் முதல் வரியின் தொடக்கத்திற்கும் இடையே உள்ள இடைவெளியைக் குறிக்கிறது. இது text-indent பண்பு மூலம் செயல்படுத்தப்படுகிறது.

Syntax

```
text-indent: spaceValue;
```

Example

```
 h2{
      letter-spacing: 10px;
    }
  p{
      line-height: 50px;
    }
  section{
      text-indent: 40px;
   }
<article>
 <h2>CSS Kerning Property</h2>
  <p>
    Line Height Property is used to define the leading value of text
in a web page.
   </p>
   <section>
    Specific the space betweeen left margin and the text
   </section>
</article>
```

Output

C S S   K e r n i n g   P r o p e r t y

Line Height Property is used to define the leading value of text in a web page.

Specific the space betweeen left margin and the text

CSS Comments

Comments ஆதாரக் குறியீட்டிற்கான ஆவணங்களை வழங்கப் பயன்படுகின்றன. இது உலாவியில் காண்பிக்கப்படாது. CSS comments <style> டேக்கிற்குள் வைக்கப்படுகின்றன.

Syntax

```
/*This is a comment*/
```

CSS Colors

CSS இல், நிறங்கள் HTML ஆவணங்களை ஸ்டைலிங் செய்வதில் முக்கிய பங்கு வகிக்கின்றன. நிறங்கள் பின்வருவனவற்றைப் பயன்படுத்தி குறிப்பிடப்படலாம்:

- Color Names
- RGB, RGBA
- HEX
- HSL, HSLA

Color Names

இதில் முன் வரையறுக்கப்பட்ட பெயர்களைப் பயன்படுத்தி வண்ணங்கள் குறிப்பிடப்படுகின்றன. உதாரணமாக Blue, Coral, Consilk, YellowGreen, etc.

Example

```
h1{
        color:blue;
 }
```

RGB, RGBA

இதில் சிவப்பு, பச்சை மற்றும் நீல மதிப்புகளின் கலவையாக நிறம் கொடுக்கப்பட்டுள்ளது. ஒவ்வொரு வண்ண மதிப்புகளும் 0-255 என்ற எண் வரம்பில் கொடுக்கப்பட்டுள்ளன

Syntax

```
rbg(red, green, blue);
```

Example

```
h1{
        color: rgb(255,0,0);
 }
```

RGBA ஆனது ஆல்பா சேனலைச் சேர்க்கிறது, இது வண்ணத்திற்கான ஒளிபுகாநிலையைக் குறிப்பிடுகிறது. ஆல்பா மதிப்பு 0.0 முதல் 1.0 வரையில் கொடுக்கப்பட்டுள்ளது. 0.0 முழு வெளிப்படைத்தன்மையைக் குறிப்பிடுகிறது மற்றும் 1.0 எந்த வெளிப்படைத்தன்மையையும் குறிப்பிடவில்லை.

Syntax

```
rgba(red, green, blue, alpha)
```

Example

```
h1{
        color: rgba(255,0,0,0.5);
 }
```

HEX value

இதில் 00 முதல் FF வரையிலான ஹெக்ஸாடெசிமல் மதிப்பாக நிறம் குறிப்பிடப்பட்டுள்ளது. இது 6 இலக்க மதிப்பாகும், இதில் முதல் 2 சிவப்பு நிற மதிப்பையும், அடுத்த 2 பச்சை நிற மதிப்பையும், கடைசி 2 நீல நிற மதிப்பையும் குறிப்பிடுகிறது.

Syntax

```
#RRGGBB
```

Example

```
h1{
        color:#00FF00;/*Green Color*/
 }
```

HSL, HSLA

HSL (Hue, Saturation, Lightness) வண்ண மாதிரியானது, வண்ணங்களை மனிதர்களுக்கு மிகவும் உள்ளுணர்வுபூர்வமாக குறிப்பிடுகிறது.

- சாயல் **(Hue):** வண்ணச் சக்கரத்தில் ஒரு குறிப்பிட்ட புள்ளியைக் குறிக்கிறது. 0 முதல் 360 வரை உள்ள ஒரு டிகிரி மதிப்பு. 0 சிவப்பு, 120 பச்சை, 240 நீலம்.
- செறிவு **(Saturation):** வண்ணத்தின் தூய்மையைக் குறிக்கிறது. 0% சாம்பல், 100% முழு நிறம்.
- லேசான தன்மை **(Lightness):** வண்ணத்தின் ஒளியைக் குறிக்கிறது. 0% கருப்பு, 100% வெள்ளை.

இந்த மூன்று மதிப்புகளை மாற்றி, பல்வேறு வண்ணங்களை உருவாக்கலாம். HSL வண்ணத் தேர்வு கருவிகள் அல்லது வண்ணத் தட்டுகளை உருவாக்கும் போது, வண்ணங்களை மிகவும் உள்ளுணர்வுபூர்வமாக கையாள வேண்டிய சூழ்நிலைகளில் HSL பெரும்பாலும் பயன்படுத்தப்படுகிறது.

Syntax

```
hsl(hue, saturation, lightness)
```

Example

```
h1{
        color:hsl(0,100,50);
 }
```

HSLA வண்ணத்திற்கு ஆல்பா மதிப்பைச் சேர்க்கிறது. ஆல்பா மதிப்பு 0.0 முதல் 1.0 வரம்பில் கொடுக்கப்படுகிறது. 0.0 முழுமையான வெளிப்படைத்தன்மையைக் குறிக்கிறது, 1.0 வெளிப்படைத்தன்மையைக் குறிக்கிறது.

Syntax

```
hsla(hue, saturation, lightness, alpha)
```

Example

```
h1{
    color:hsla(0,100,50,0.5);
}
```

Background

Background Property CSS இல் ஒரு உறுப்பின் பின்னணி விளைவுகளை வரையறுக்கப் பயன்படுகிறது. சில முக்கியமான பின்னணி பண்புகள்:

background-color: இது ஒரு உறுப்பின் பின்னணி நிறத்தை அமைக்கப் பயன்படுகிறது.

Syntax

```
background-color: colorvalue:
```

background-image: இது ஒரு உறுப்பின் பின்னணி படத்தை அமைக்கப் பயன்படுகிறது.

Syntax

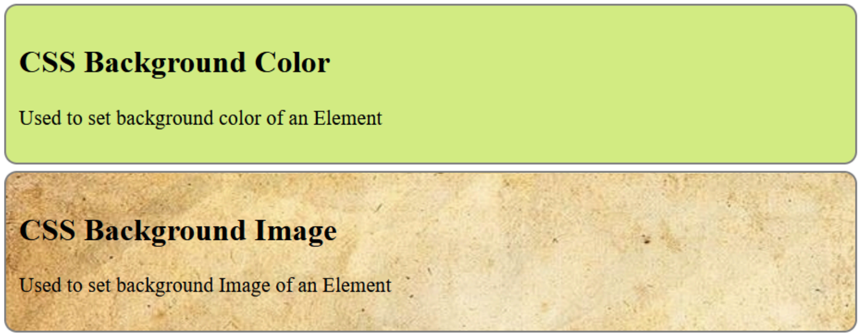
```
background-image: url(“Image URL”);
```

Example

```
article{
    width: auto;
    border: 2px solid gray;
    padding: 10px;
    border-radius: 10px;
    margin:5px;
    background-color: #d2ed83;
    }
div{
    width: auto;
    border: 2px solid gray;
    padding: 10px;
    border-radius: 10px;
    margin:5px;
    background-image: url("oldnote.jpg");
    background-color: #DFD2A6;
    }

<article>
  <h2>CSS Background Color</h2>
  <p>
    Used to set background color of an Element
  </p>
</article>
<div>
  <section>
    <h2>CSS Background Image</h2>
    <p>
     Used to set background Image of an Element
    </p>
  </section>
</div>
```

Output



background-position: இது உறுப்புக்குள் படத்தின் நிலையை அமைக்கப் பயன்படுகிறது. நிலை top left right bottom (அல்லது) பிக்சல் நிலை என வழங்கப்படுகிறது.

Syntax

```
background-position: right bottom;
```

background-repeat: ஒரு படம் மீண்டும் மீண்டும் காண்பிக்கப்பட வேண்டிய திசையை வரையறுக்க இது பயன்படுகிறது. இந்த சொத்துக்கு கொடுக்கப்பட்ட மதிப்புகள் repeat, repeat-x, repeat-y and no-repeat ஆகும்

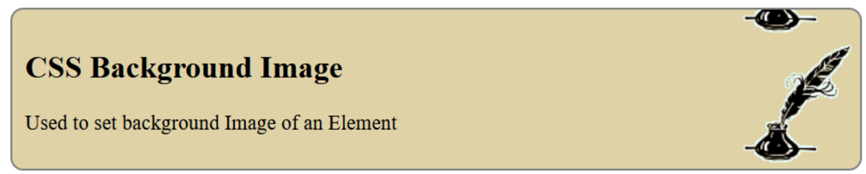
Syntax

```
background-repeat: repeat-y;
```

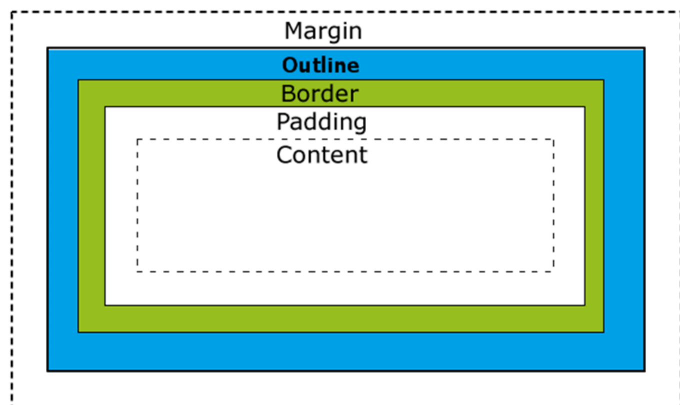
Example

```
div{
    width: auto;
    border: 2px solid gray;
    padding: 10px;
    border-radius: 10px;
    margin:5px;
    background-image: url("pen1.png");
    background-position: right bottom;
    background-repeat: repeat-y;
    background-color: #DFD2A6;
}
<div>
    <section>
        <h2>CSS Background Image</h2>
        <p>
         Used to set background Image of an Element
        </p>
    </section>
</div>
```

Output



CSS Box Model



Borders

ஒரு உறுப்பைக் குறிக்கும் பெட்டியின் எல்லையைக் குறிப்பிட CSS பார்டர் பண்பு பயன்படுத்தப்படுகிறது. சில முக்கியமான எல்லை பண்புகள்

border-color: எல்லையின் நிறத்தை அமைக்க இது பயன்படுகிறது

Syntax

```
border-color: red;
```

border-style: எல்லைக் கோடு பாணியை அமைக்க இது பயன்படுகிறது.. Examples of line style are dotted, dashed, solid, double, none, hidden, etc.

Syntax

```
border-style: dotted;
```

border-width: எல்லைக் கோட்டின் அகலத்தை அமைக்க இது பயன்படுகிறது.

Syntax

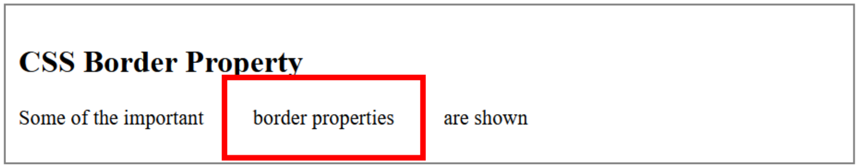
```
border-width: 3px;
```

Example

```
span{
   border-width: 5px;
   border-style: solid;
   border-color: RED;
   margin: 10px;
   padding: 20px;
}

<article>
   <h2>CSS Border Property</h2>
   <p>
      Some of the important <span>border properties</span>
are shown
   </p>
</article>
```

Output



Outline

CSS இல், ஒரு உறுப்பைச் சுற்றி காணக்கூடிய வெளிப்புறத்தை உருவாக்க அவுட்லைன் சொத்து பயன்படுத்தப்படுகிறது. அவுட்லைன் சொத்து தனிமத்தின் எல்லையில் இருந்து தனித்தனியாக உள்ளது மற்றும் உறுப்பின் தளவமைப்பை பாதிக்காது. அவுட்லைன் பண்புகள் சில

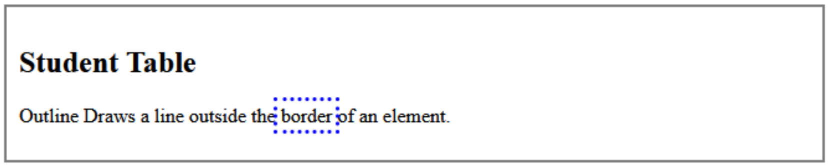
- **outline-style**: வரி நடையை அமைக்க பயன்படுகிறது
- **outline-color**: வரியின் நிறத்தை அமைக்க பயன்படுகிறது
- **outline-width**: கோட்டின் அகலத்தை அமைக்க பயன்படுகிறது
- **outline-offset**: அவுட்லைன் மற்றும் பார்டர் இடையே இடைவெளி அமைக்க பயன்படுகிறது

Example

```
span{
   outline-style: dotted;
   outline-color: blue;
   outline-width: 3px;
   outline-offset: 3px;
```

```
}
<article>
  <h2>Student Table</h2>
  <p>Outline Draws a line outside the <span>border</span> of
     an element
 . </p>
</article>
```

Output



Margin

CSS margin பண்பு ஒரு உறுப்புக்கான மார்ஜின் இடைவெளியை அமைக்கப் பயன்படுகிறது. இது அடுத்தடுத்த உறுப்புகளுக்கு இடையே உள்ள இடைவெளியை அமைக்கிறது. Margin-top, margin-right, margin-bottom மற்றும் margin-right ஆகியவற்றைப் பயன்படுத்தி தனித்தனியாக அல்லது margin-ஐப் பயன்படுத்தி நான்கு பக்கங்களுக்கும் அமைக்கலாம்.

Syntax

```
margin: 10px;
```

Example

Same as border Example

Padding

CSS padding பண்பு ஒரு உறுப்பின் எல்லைக்குள் அதைச் சுற்றியுள்ள இடைவெளியை அமைக்கப் பயன்படுகிறது. Padding-top, padding-right, padding-bottom மற்றும் padding-left ஆகியவற்றைப் பயன்படுத்தி தனித்தனியாக அல்லது padding-ஐப் பயன்படுத்தி நான்கு பக்கங்களுக்கு அமைக்கலாம்.

Syntax

```
padding: 10px;
```

Example

Same as border Example

List

HTML பட்டியல் உறுப்புகளின் display பண்பு CSS மூலம் மாற்றப்படலாம். இது list-style-type பண்பு மூலம் செய்யப்படுகிறது. வரிசையற்ற பட்டியல் பாணிகள் circle, square மற்றும் disc போன்றவற்றைக் காட்டலாம். வரிசைப்படுத்தப்பட்ட பட்டியல் பாணிகள் decimal, upper-alpha போன்றவற்றைக் காட்டலாம். படத்தை பட்டியல் பாணி list-style-image பண்பு மூலம் காட்டலாம்.

Example

```
ul{
      list-style-type: square;
      #list-style-image: url("dept.png");
}
ol{
      list-style-type: upper-alpha;
}
<section>
```

```
        <h4>Style Sheet Types</h4>
        <ul>
        <li>Embedded</li>
        <li>External</li>
        <li>In-Line</li>
        </ul>
        <ol>
        <li>HTML5</li>
        <li>CSS3</li>
        <li>JavaScript</li>
        </ol>
</section>
```

Output

Tables

HTML tableகள் இயல்பாகவே எல்லை வரிகள் மற்றும் பிற பாணிகளைக் காண்பிக்காது. அட்டவணிகளுக்கு எல்லைகள் மற்றும் பிற பாணிகளைக் காண்பிக்க CSS பயன்படுத்தப்படுகிறது. tableகளுடன் தொடர்புடைய மிகவும் அடிக்கடி பயன்படுத்தப்படும் சில CSS பண்புகள்

- **border:** இந்த பண்பு அட்டவணிகளுக்கான எல்லைகளை அமைக்கப் பயன்படுகிறது.
- **border-collapse:** இந்த பண்பு border பண்பு மூலம் உருவாக்கப்பட்ட இரட்டை எல்லைகளை ஒற்றை எல்லையாக இணைக்கப் பயன்படுகிறது.
- **width, height:** இந்த இரண்டு பண்புகளும் அட்டவணையின் அகலம் மற்றும் உயரத்தை அமைக்கப் பயன்படுகிறது.
- **text-align:** இந்த பண்பு அட்டவணையில் உள்ள உரையின் சீரமைப்பை அமைக்கப் பயன்படுகிறது.

Example

```
table, th, td{
   border: 2px dotted;
}
table{
   border-collapse: collapse;
   width: 100%;
   height: 100px;
}
th{
   text-align: right;
}
<article>
   <h2>Student Table</h2>
   <table>
    <tr>
      <th>S.No.</th>
      <th>Name</th>
    </tr>
    <tr>
```

```
        <td>1</td>
        <td>Arvind</td>
      </tr>
      <tr>
        <td>2</td>
        <td>Bashir</td>
      </tr>
      <tr>
        <td>3</td>
        <td>Charan</td>
      </tr>
    </table>
  </article>
```

Output

Student Table

S.No.	Name
1	Arvind
2	Bashir
3	Charan

Table Special Effects

CSS ஐப் பயன்படுத்தி அட்டவணைக்கு சில சிறப்பு விளைவுகளைச் சேர்க்கலாம், எடுத்துக்காட்டாக, மவுஸ் அதன் மேல் வைக்கப்படும் போது ஒரு வரியை முன்னிலைப்படுத்தி, nth-child() தேர்ந்தெடுப்பைப் பயன்படுத்தி ஸ்ட்ரிப் செய்யப்பட்ட அட்டவணிகளை உருவாக்குதல்.

Example

```
table, th, td{
   border: 2px dotted;
}
table{
   border-collapse: collapse;
   width: 100%;
}
tr:nth-child(even){
   background-color: grey;
}
tr:hover{
   text-align: right;
   background-color: lightblue;
}
<article>
   <h2>Student Table</h2>
   <table>
     <tr>
        <th>S.No.</th>
        <th>Name</th>
     </tr>
     <tr>
        <td>1</td>
        <td>Arvind</td>
     </tr>
     <tr>
        <td>2</td>
        <td>Bashir</td>
     </tr>
     <tr>
```

```
          <td>3</td>
          <td>Charan</td>
        </tr>
      </table>
  </article>
```

Output

Student Table

S.No.	Name
1	Arvind
2	Bashir
3	Charan

Forms

HTML படிவங்கள் பயனாளர்களிடமிருந்து உள்ளீட்டைப் பெறப் பயன்படுகின்றன. படிவ உறுப்புகள் CSS மூலம் ஸ்டைலிங் செய்யப்படலாம். இதற்கு உறுப்பு தேர்ந்தெடுப்பாளர் சிறிய மாற்றங்களுடன் பயன்படுத்தப்படுகிறது. படிவங்களை வடிவமைக்கப் பயன்படுத்தப்படும் திருத்தப்பட்ட தேர்ந்தெடுப்பாளர்கள்:

- **input[type=text]**: உரை உள்ளீட்டு புலங்களைத் தேர்ந்தெடுக்கிறது.
- **input[type=button]**: பொத்தான் உள்ளீட்டு புலங்களைத் தேர்ந்தெடுக்கிறது.
- **select**: தேர்வு பட்டியல் உள்ளீட்டு புலங்களைத் தேர்ந்தெடுக்கிறது.

பின்வரும் CSS பண்புகள் படிவ உறுப்புகளுக்குப் பயன்படுத்தப்படலாம்:

- Width
- Padding
- Border
- Margin
- Color
- Hover, focus
- Icon/Image

Example

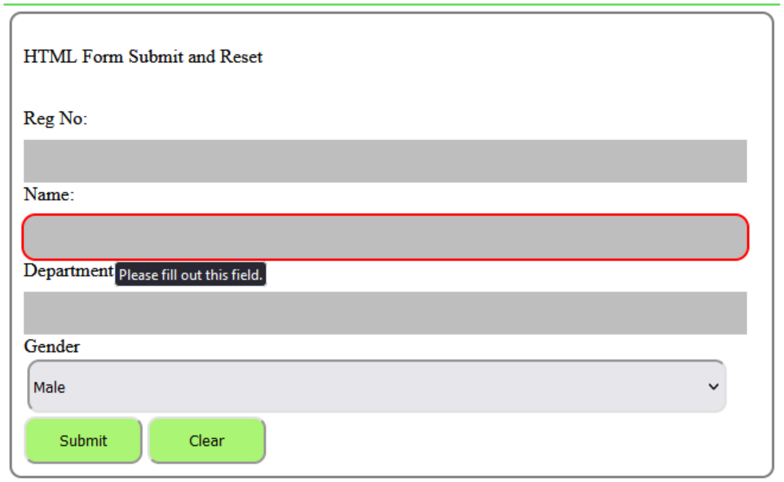
```
input[type=text]{
   display: block;
   width: 95%;
   padding: 10px;
   margin-top: 10px;
   border: none;
   background-color: rgba(127,127,127,0.5);
}

input[type=text]:hover{
   outline: 2px solid red;
   border-radius: 10px;
}

input[type=text]:focus{
   background-color:white;
   color: black;
   outline:none;
   border: none;
   border-radius: 10px;
   border-bottom: 2px solid red;
}
input[type=text].dept:focus{
   background-image:url("dept.png");
   background-position: 10px 5px;
   background-repeat: no-repeat;
```

```
    padding-left: 50px;
  }
  select{
    display: block;
    width: 95%;
    height: 45px;
    padding: 3px;
    margin: 3px;
    border-radius: 10px;
  }
  select:hover{
    outline: 2px solid red;
    border-radius: 10px;
  }
  input[type=submit], input[type=reset]{
    width: 100px;
    height: 40px;
    border-radius: 10px;
    padding: 3px;
    background-color: #aef777;
  }
  input[type=submit]:hover, input[type=reset]:hover{
    outline: 2px solid red;
    border-radius: 10px;
    color: #aef777;
    background-color: grey;
  }
  <article>
    <p>HTML Form Submit and Reset</p><br>
    <form id="myform">
      <div>
        <label for="regno" >Reg No: </label>
        <input type="text" id="regno" required>
        </div>
        <div>
        <label for="txtname" >Name: </label>
        <input type="text" id="txtname" required>
        </div>
        <div>
        <label for="txtdpt" >Department: </label>
        <input class="dept" type="text" id="txtdpt">
        </div>
        <div>
        <label for="selgndr">Gender</label>
        <select id="selgndr">
         <option value="M">Male</option>
         <option value="F">Female</option>
        </select>
        </div>
        <div>
        <input type="submit" value="Submit">
        <input type="reset" value="Clear">
        </div>
    </form>
  </article>
```

Output



Links

HTML linkகள் CSS மூலம் ஸ்டைலிங் செய்யப்படலாம். ஒரு link நான்கு நிலைகளைக் கொண்டுள்ளது: சாதாரண பார்வையிடப்படாத இணைப்பு, பார்வையிடப்பட்ட இணைப்பு, மவுஸ் மீது உள்ள இணைப்பு நிலை மற்றும் செயலில் உள்ள இணைப்பு. இந்த நான்கு நிலைகளும் CSS இல் ஆங்கர் டேக்கின் சூடோ கிளாஸ் மூலம் ஸ்டைலிங் செய்யப்படலாம், எடுத்துக்காட்டாக:

a:link
a:visited
a: hover
a:active

இதேபோல் இணைப்பு பாணியை அகற்றி, "**none**" மதிப்புடன் சாதாரண **text-decoration** பண்புக்கூறாகக் காட்டவும் பயன்படுத்தப்படுகிறது

Example

```
a:link{
        color: RED;
        font-family: monospace;
        font-size: 20px;
}
a:hover{
        text-shadow: 5px 5px 5px blue, 3px 3px 6px yellow;
}
a:visited{
        color: grey;
        font-family: monospace;
        font-size: 10px;
}
a:active{
        color: blue;
        font-family: monospace;
        font-size: 30px;
}
<article>
        <h2>Link Styles</h2>
        <section>
        <a  href="https:\\www.google.com"  taget="_blank">Sample
Link</a>
        </section>
</article>
```

Output

4.2 Advanced CSS3

CSS Rounded Corners

CSS இல், வட்டமான மூலைகள் boxes, buttons, மற்றும் divs போன்ற கூறுகளுக்கு border-radius பண்பு மூலம் உருவாக்கப்படலாம். இந்த பண்பு ஒரு கூறு மூலைகளின் வளைவை கட்டுப்படுத்த உதவுகிறது, இது வடிவமைப்புக்கு மிகவும் காட்சிப்படுத்தும் மற்றும் நவீன தோற்றத்தை தருகிறது.

- **border-radius**: இது ஒவ்வொரு மூலைக்கும் 4 மதிப்புகள் அல்லது விட்டக மதிப்புகளை குறிக்கும் 2 மதிப்புகள் அல்லது நான்கு மூலைகளுக்கும் பயன்படுத்தப்படும் 1 மதிப்பு ஆகியவற்றைக் கொண்டிருக்கலாம்.

Syntax

border-radius: 10px; */All four corners/*

Border Images

CSS border images என்பது HTML கூறுகளைச் சுற்றி அலங்கார எல்லைகளை உருவாக்க ஒரு சக்திவாய்ந்த கருவியாகும். இவை வலை உள்ளடக்கத்தின் காட்சி ஈர்ப்பை அதிகரிக்க எளிய, திட நிற கோடுகளுக்குப் பதிலாக படங்களைப் பயன்படுத்த அனுமதிக்கின்றன.

CSS border-image shorthand notation என்பது பட எல்லைகளுடன் தொடர்புடைய பல பண்புகளை வரையறுக்க ஒரு சுருக்கமான வழியாகும். இது படங்களுடன் அலங்கார எல்லைகளை உருவாக்கும் செயல்முறையை எளிமைப்படுத்துகிறது. சுருக்க குறியீடு பின்வரும் பண்புகளை உள்ளடக்கியது:

1. **border-image-source:** இந்த பண்பு எல்லையாகப் பயன்படுத்தப்படும் படத்திற்கான பாதையைக் குறிப்பிடுகிறது. இது பொதுவாக படக் கோப்புக்குச் சுட்டிக்காட்டும் ஒரு URL ஆகும்.
2. **border-image-slice:** இந்த பண்பு படத்தை எல்லை துண்டுகளாகப் பயன்படுத்தப்படும் பிரிவுகளாக வெட்டப்படும் விதத்தை வரையறுக்கிறது. நீங்கள் விரும்பிய விளைவை உருவாக்க இந்த துண்டுகளின் அளவு மற்றும் பரிமாணங்களை கட்டுப்படுத்தலாம்.
3. **border-image-width:** இந்த பண்பு எல்லைப் பட துண்டுகளின் அகலத்தை தீர்மானிக்கிறது, அவை கூறுக்குச் சுற்றி எவ்வாறு காட்சிப்படுத்தப்படும் என்பதை குறிப்பிடுகிறது.
4. **border-image-outset:** இந்த பண்பு எல்லை கூறு பெட்டியின் தாண்டி எவ்வளவு தொலைவு நீட்டிக்கும் என்பதை குறிப்பிடுகிறது, கூறுக்கும் எல்லைப் படத்திற்கும் இடையே உள்ள இடைவெளியை கட்டுப்படுத்துகிறது.
5. **border-image-repeat:** இந்த பண்பு எல்லைப் படத்தின் டைலிங் அல்லது repetition நடத்தையை அமைக்கிறது, நீங்கள் வித்தியாசமான விளைவுகளை அடைய

நீட்டல், மீண்டும் அல்லது வட்டம் போன்ற மதிப்புகளிலிருந்து தேர்வு செய்ய அனுமதிக்கிறது.

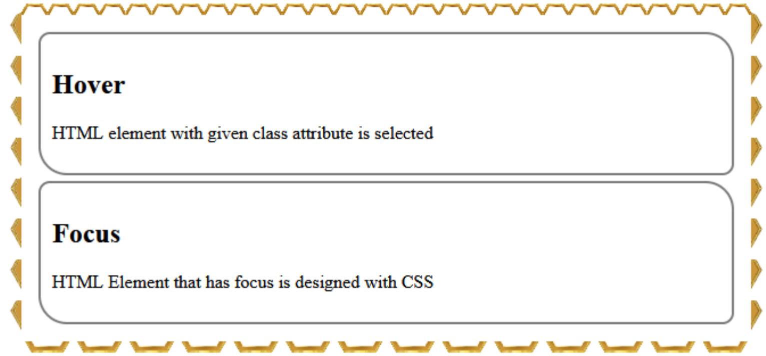
Syntax

```
border-image: url('border-image.png') 20 10px 5px round;
```

Example

```
.cont{
   width: auto;
   border: 10px solid transparent;
   padding: 10px;
   margin:5px;
   background-color: transparent;
   border-image: url("bord4.png") 40% 10% 20% 10% round;
}
article{
   width: auto;
   border: 2px solid gray;
   padding: 10px;
   border-radius: 10px 25px 10px 25px;
   margin:5px;
}
<div class="cont">
<article>
   <h2>Hover</h2>
   <p>HTML element with given class attribute is selected</p>
</article>
<article>
   <h2>Focus</h2>
   <p>HTML Element that has focus is designed with CSS</p>
</article>
</div>
```

Output



CSS Background Image

CSS உங்களுக்கு HTML elementகளுக்கு பின்னணி படங்களை அமைக்க அனுமதிக்கிறது. சில முக்கியமான CSS background image பண்புகள் இதில் அடங்கும்:

- **background-image**: இந்த பண்பு ஒரு உறுப்பின் பின்னணியாக பல படங்களை வைக்க அனுமதிக்கிறது.
- **background-size**: பின்னணி படத்தின் அளவை அமைக்க இந்த பண்பு பயன்படுத்தப்படுகிறது. இது பிக்சல், கொண்டிருக்கும் மற்றும் கவர் மதிப்புகளை எடுக்கும்.
- **background-origin**: பாக்ஸ் சொத்தை பயன்படுத்தி படத்தை பின்னணியில் வைக்க இந்த பண்பு பயன்படுத்தப்படுகிறது. இது padding-box, border-box, content-box மதிப்புகளை எடுக்கும்.
- **background clip**: உறுப்பின் பெட்டியுடன் தொடர்புடைய ஒரு உறுப்பின் பின்னணி எவ்வாறு வெட்டப்படுகிறது அல்லது வர்ணம் பூசப்படுகிறது என்பதைக் கட்டுப்படுத்த இந்தப் பண்பு பயன்படுத்தப்படுகிறது. இது padding-box, border-box, content-box மதிப்புகளை எடுக்கும்.

Color Keywords

CSS இல், நிறச் சொற்கள் குறிப்பிட்ட நிறங்களை பிரதிநிதிப்படுத்தும் முன்கூட்டியே வரையறுக்கப்பட்ட சொற்களாகும். இந்த சொற்கள் துல்லியமான எண்கணித நிற மதிப்புகளை குறிப்பிடாமல் நிறங்களை அமைக்க எளிதாக்குகின்றன. நிற பெயர்களுக்கு மேலதிகமாக சில முக்கியமான நிற சொற்கள் இதில் அடங்கும்

- **transparent**: இந்த சொல் முற்றிலும் வெளிப்படையான நிறத்தைக் குறிக்கிறது.
- **inherit**: இந்த சொல் ஒரு கூறு அதன் பெற்றோர் கூறுகிடமிருந்து அதன் நிறத்தை மரபுரிமை பெற அனுமதிக்கிறது.
- **currentcolor**: இந்தச் சொல் elementன் வண்ணப் பண்புகளைப் பயன்படுத்துகிறது.

Gradient

CSS இல் Gradients ஒரு குறிப்பிட்ட கூறுக்குள் பல நிறங்கள் அல்லது நிற நிறுத்தங்களுக்கு இடையே ஒரு மென்மையான மாற்றத்தை உருவாக்க ஒரு வழியாகும். அவை பின்னணி, எல்லைகள் மற்றும் உரை போன்ற பல்வேறு வடிவமைப்பு கூறுகளுக்குப் பயன்படுத்தப்படுகின்றன.

Types of Gradients are

- **linear-gradient**: Linear gradients ஒரு நேர் கோட்டில் வண்ண மாற்றத்தை உருவாக்குகிறது..
- **radial-gradient**: Radial gradient ஒரு மைய புள்ளியிலிருந்து வெளிப்புறமாக ஒரு வட்ட வடிவத்தில் வண்ண மாற்றத்தை உருவாக்குகின்றன.
- **conic-gradient**: Conical gradients ஒரு கூம்பு வடிவத்தில் ஒரு மைய புள்ளியிலிருந்து வெளிப்புறமாக ஒரு வண்ண மாற்றத்தை உருவாக்குகிறது.

Example

```
background: linear-gradient(to right, #FF0000 0%, #FFFF00 50%, #00FF00 100%);

background: conic-gradient(at 50% 50%, red 0%, yellow 25%, green 50%, blue 100%);

background: radial-gradient(circle, #FF0000, #0000FF);
```

Shadow Effects

CSS இல், நிழல் விளைவுகளை கூறுகளுக்குப் பயன்படுத்தி ஆழம் மற்றும் பரிமாணத்தை உருவாக்கலாம். இந்த விளைவுகள் பொதுவாக ஒரு மேற்பரப்பில் நிழல்களை வீசும் பொருள்களின் தோற்றத்தை உருவகப்படுத்த பயன்படுத்தப்படுகின்றன. சில முக்கிய நிழல் பண்புகள் இதில் அடங்கும்

- **text-shadow**: இது குறிப்பாக கூறுகளுக்குள் text நிழலாட வைக்கப் பயன்படுகிறது.
- **box-shadow**: இது பொதுவாக பெட்டிகள் அல்லது கொள்கலன்களுக்கு நிழல்களைச் சேர்க்கப் பயன்படுகிறது.

Syntax

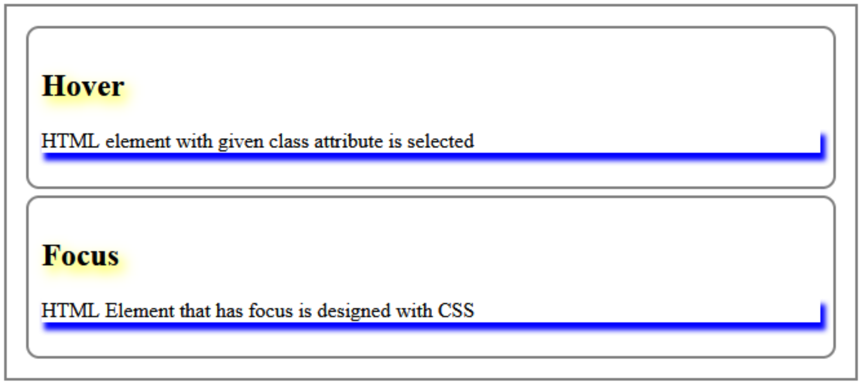
```
text-shadow: x y blur color;
box-shadow: x y blur spread color <inset>;
```

Example

```
h1, h2{
   text-shadow: 4px 5px 10px yellow;
}
p{
```

```
    box-shadow: 3px 5px 3px 1px blue inset;
}
<article>
    <h2>Hover</h2>
    <p>HTML element with given class attribute is selected</p>
</article>
<article>
    <h2>Focus</h2>
    <p>HTML Element that has focus is designed with CSS</p>
</article>
```

Output



Web Fonts

வலை எழுத்துருக்கள் என்பது வலைப்பக்கங்களில் பயன்படுத்தக்கூடிய தனிப்பயன் எழுத்துருக்களைக் குறிக்கின்றன. அவை பயனர்களின் சாதனங்களில் கிடைக்கும் நிலையான எழுத்துருக்களுக்கு மட்டுப்படுத்தப்படவில்லை. இந்த எழுத்துருக்கள் சேவையகத்திலிருந்து பதிவிறக்கம் செய்யப்படுகின்றன. வலை எழுத்துருக்கள் @font-face விதியைப் பயன்படுத்தி ஒரு வலைப்பக்கத்தில் சேர்க்கப்பட்டுள்ளன. இது ttf, OTF மற்றும் woff போன்ற எழுத்துரு வகைகளை ஏற்றுக்கொள்கிறது.

Example

```
@font-face{
    font-family: myTamilFont;
    src: url(Font/Alankaram.otf);
}
h1{
    font-family: myTamilFont;
}
<header>
    <h1>jkpo;</h1>
</header>
```

Output

Text Effects

CSS இல், வலைப்பக்கங்களில் உரை உள்ளடக்கத்தின் தோற்றத்தை மேம்படுத்த பல்வேறு உரை விளைவுகளைப் பயன்படுத்தலாம். உரை விளைவுகள் கண்கவர் அச்சுக்கலை உருவாக்க அனுமதிக்கின்றன மற்றும் உங்கள் வலைத்தளத்தின் காட்சி முறையீட்டை மேம்படுத்துகின்றன. உரை விளைவு பண்புகளில் சில

- **text-overflow**: இந்த பண்பு அதன் கொள்கலனின் எல்லைகளைத் தாண்டி விரிவடையும் போது உரை எவ்வாறு செயல்படுகிறது என்பதைக் கட்டுப்படுத்துகிறது. இது கிளிப் மற்றும் எலிப்சிஸ் மதிப்புகளை எடுக்கும்.
- **word-wrap**: இந்த பண்பு நீண்ட வார்த்தைகள் அல்லது எழுத்துக்களின் தொடர்கள் அவற்றின் கொள்ளளவு

கூறுகளின் எல்லைகளை தாண்டும் போது எவ்வாறு கையாளப்படுகின்றன என்பதை கட்டுப்படுத்த பயன்படுகிறது. இது break-word என்ற மதிப்பை எடுக்கிறது.

- **writing-mode**: இந்த பண்பு உரையின் திசையை வரையறுக்கிறது, இது ஒரு கூறின் உள்ளே உரை உள்ளடக்கத்தின் ஓட்டம் மற்றும் திசையை கட்டுப்படுத்துகிறது. இது horizontal-tb, vertical-lr மற்றும் vertical-rl என்ற மதிப்புகளை எடுக்கிறது.

Example

```
div{
   width: 50px;
   overflow: hidden;
   border: 2px solid black;
   //text-overflow: clip;
   word-wrap: break-word;
   //writing-mode: vertical-lr;
}
<article>
   <h2>Hover/Focus</h2>
   <div>HTML element with given class attributeisselected.</div>
</article>
```

Output

2D Transforms

இந்த 2D உருமாற்றங்கள் CSS இல் HTML கூறுகளின் நிலை, சுழற்சி, அளவு மற்றும் சாய்வை இரு பரிமாண இடைவெளியில் மாற்றுவிக்க அனுமதிக்கின்றன. 2D உருமாற்றத்தை நிறைவேற்ற **transform** பண்பு பயன்படுகிறது. வழங்கப்படும் மதிப்புகளின் அடிப்படையில் வெவ்வேறு 2D உருமாற்றங்கள் செய்யப்படுகின்றன.

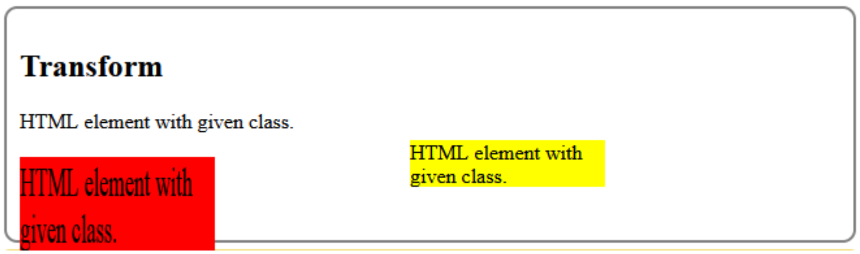
- **translate(x,y)**: இந்த செயல்பாடு ஒரு கூறை கிடைமட்ட மற்றும் நிலைக்குத்து அச்சுகளில் நகர்த்துகிறது.
- **rotate(deg)**: இந்த செயல்பாடு ஒரு கூறை அதன் மையத்தில் குறிப்பிட்ட கோணத்தில் சுழற்றுகிறது.
- **scale(x/y); [scaleX/scaleY]**: scaleX செயல்பாடு ஒரு கூறை x-அச்சில் அளவை மாற்றுகிறது மற்றும் scaleY செயல்பாடு ஒரு கூறை y-அச்சில் அளவை மாற்றுகிறது.
- **skew(xdeg, ydeg); [skewX/skewY]**: இந்த செயல்பாடுகள் ஒரு கூறை குறிப்பிட்ட கோணத்தில் சாய்வு செய்கின்றன. skewX செயல்பாடு ஒரு கூறை கிடைமட்ட திசையில் சாய்வு செய்கிறது, அதேபோல் skewY செயல்பாடு ஒரு கூறை செங்குத்து திசையில் சாய்வு செய்கிறது.

Example

```
.div1{
   width: 150px;
   background-color: yellow;
   transform: translate(300px,5px);
   //transform: rotate(45deg);
}
.div3{
```

```
    width: 150px;
    background-color: Red;
    transform: scaleY(2);
 }
 <article>
    <h2>Transform</h2>
    <div>HTML element with given class.</div>
    <div class="div1">HTML element with given class.</div>
    <div class="div3">HTML element with given class.</div>
 </article>
```

Output



3D Transforms

3D மாற்றங்கள் CSS இல் HTML கூறுகளின் நிலை, சுழற்சி மற்றும் அளவையை மூன்று பரிமாண இடைவெளியில் கையாள்வதை அனுமதிக்கின்றன. transform பண்பு இந்த 3D மாற்றங்களை நிறைவேற்றுவதற்கு பயன்படுகிறது.

- **Scale**: CSS scaleX(), scaleY(), scaleZ() செயல்பாடுகள் 3D இடத்தில் x, y மற்றும் z அச்சுகளுடன் ஒரு உறுப்பை அளவிடுவதற்குப் பயன்படுத்தப்படுகின்றன.
- **Rotate**: CSS rotateX(), rotateY(), மற்றும் rotateZ() செயல்பாடுகள் 3D இடத்தில் உள்ள உறுப்புகளுக்கு சுழற்சி மாற்றங்களைப் பயன்படுத்த பயன்படுகிறது.

Buttons

<button> குறிச்சொல்லைப் பயன்படுத்தி உருவாக்கப்பட்ட HTML பொத்தானின் தோற்றத்தை பின்வரும் CSS பண்புகளைப் பயன்படுத்தி மாற்றியமைக்க முடியும்.

- border
- border-radius
- background-color
- color
- hover

Example

```
 .danger{
    background-color: rgba(0,125,0,0.5);
    font-size: 24px;
    border-radius: 10px;
 }
 .danger:hover{
    background-color: rgba(123,0,0,0.5);
 }
 <div>
        <button class="danger">
        <span class="material-icons">edit</span>
        Edit
        </button>
 </div>
```

Output

Counters

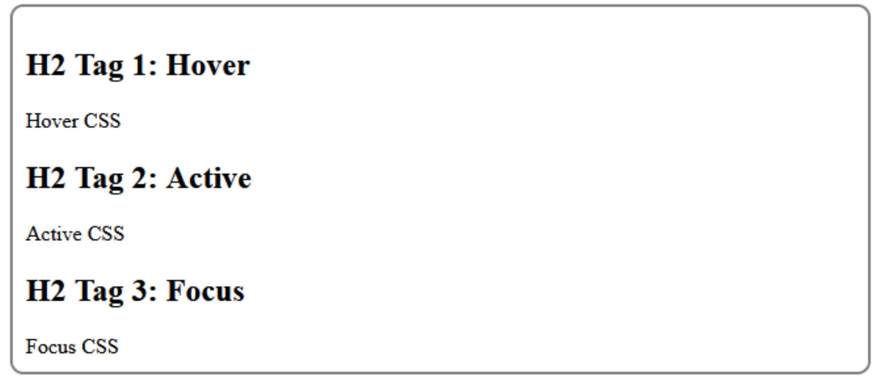
CSS counterகள் ஒழுங்குபடுத்தப்பட்ட பட்டியல்களில் அல்லது பிற கூறுகளில் தானாகவே எண்ணிக்கையை வழங்கும் ஒரு வழி. இது பின்வரும் பண்புகளைக் கொண்டுள்ளது:

- **counter-reset**: இது தானியங்கி எண்ணை வழங்குவதற்கான கவுண்டரை உருவாக்குகிறது
- **counter-increment**: எண்ணிக்கையை அதிகரிக்கும் பண்பு.
- **content**: எந்தவொரு உரை உள்ளடக்கத்தையும் சேர்க்க இது பயன்படுத்தப்படுகிறது.
- **counter()**: இது counter மதிப்பைச் சேர்க்கப் பயன்படுகிறது.

Example

```
 body{
    counter-reset: heading;
 }
 h2::before{
    counter-increment: heading;
    content: "H2 Tag " counter(heading) ": ";
 }
 <article>
    <h2>Hover</h2>
    <div>Hover CSS</div>
    <h2>Active</h2>
    <div>Active CSS</div>
    <h2>Focus</h2>
    <div>Focus CSS</div>
 </article>
```

Output



CSS ToolTip

CSS இல் டூல்டிப்கள் ஒரு பயனர் ஒரு கூறு மீது சுட்டிக்காட்டும் போது தோன்றும் தகவல் pop-up உரையாகும். இந்த டூல்டிப்கள் பொதுவாக ஒரு கூறு பற்றிய கூடுதல் தகவலை வழங்க அல்லது பயனர் அனுபவத்தை மேம்படுத்த பயன்படுத்தப்படுகின்றன. <span> டேக் டூல்டிப் உரையை உள்ளடக்க பயன்படுத்தப்படுகிறது. நிலை பண்பு டூல்டிப் உரையை நிலைநிறுத்த பயன்படுத்தப்படுகிறது.

Example

```
 .tooltip{
    position: relative;
    display:inline-block;
 }
 .tooltip .tiptext{
```

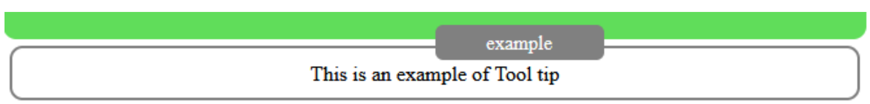
```
        visibility: hidden;
        font-size: 15px;
        width: 120px;
        background-color: grey;
        color: #fff;
        padding: 5px;
        border-radius: 6px;

        position: absolute;
        z-index: 1;
        bottom: 105%;
        left: 50%;
}

.tooltip:hover .tiptext{
    visibility: visible;
}

<article>
    <div class="tooltip">
        This is an example of Tool tip
        <span class="tiptext">example</span>
    </div>
</article>
```

Output



CSS Transition

CSS மாற்றங்கள் CSS பண்புகளை ஒரு நிலையில் இருந்து மற்றொரு நிலைக்கு மென்மையாக மாற்ற அனுமதிக்கின்றன. இந்த அனிமேஷன்கள் வலைப் பக்கங்களின் தொடர்பு மற்றும் பயனர் அனுபவத்தை மேம்படுத்த ஒரு காட்சிப்படுத்தும் மற்றும் பயனர் நட்பு வழியை வழங்குகின்றன. சில மாற்றம் பண்புகள் இதில் அடங்கும்:

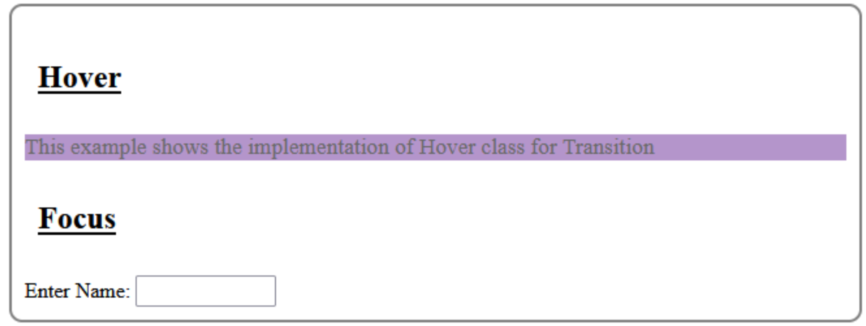
- **transition-delay**: இது ஒரு மாற்றத்தின் தொடக்கத்தை தாமதப்படுத்தும் காலத்தைக் குறிக்கிறது.
- **transition-duration**: இது ஒரு மாற்றத்தைச் செய்ய தேவையான நேரத்தைக் குறிக்கிறது.
- **transition-property**: இது மாற்றம் பொருந்த வேண்டிய CSS பண்புகளை வரையறுக்கிறது. பல பண்பு பெயர்கள் கமாவால் பிரிக்கப்பட்டு பயன்படுத்தப்படலாம். “all” என்ற சொல் அனைத்து CSS பண்புகளிலும் மாற்றங்களை செயல்படுத்தப் பயன்படுகிறது.
- **transition-timing-function**: இது கொடுக்கப்பட்ட காலத்திற்கு ஒரு மாற்றம் எவ்வாறு தொடரும் என்பதை விவரிக்கும் செயல்பாட்டை வரையறுக்கிறது. இது ease, ease-in, ease-out மற்றும் linear என்ற மதிப்புகளை எடுக்கும்.
- **transition**: இது மாற்றம் பண்புக்கான சுருக்க குறியீடாகும்.

Example

```
p{
    transition-property: font-size, color, background-color;
    transition-duration: 3s;
    transition-timing-function: linear;
}
```

```
p:hover{
   color: white;
   font-size: larger;
   background-color: indigo;
}
input{
   transition: all 1s ease-in-out;
   width: 100px;
   height: 18px;
   font-size: normal;
}
input:focus {
   width: 250px;
   height: 50px;
   font-size: larger;
   background-color: lightgray;
}
<article>
   <h1>Hover</h1>
   <p>
      This example shows the implementation of Hover class for Transition
   </p>
   <h1>Focus</h1>
   <div>
      <label for="studname">Enter Name: </label>
      <input type="text" name="studname">
   </div>
</article>
```

Output



CSS Animation

CSS அனிமேஷன்கள் வலைப்பக்கங்களில் இயக்கமான மற்றும் கண்கவர் விளைவுகளை உருவாக்க அனுமதிக்கின்றன. மாற்றங்களைக் கையாளும் ஒரு நிலையிலிருந்து இன்னொரு நிலைக்கு, அனிமேஷன்கள் keyframesகளை வரையறுக்கவும் அனிமேஷன் வரிசையில் இடைநிலை படிகளை கட்டுப்படுத்தவும் அனுமதிக்கின்றன. இது இரண்டு படிநிலை செயல்முறையாகும்.

1. முதலில், @keyframes விதியை பயன்படுத்தி அனிமேஷன் வரிசையை வரையறுக்க வேண்டும்.
2. பின்னர், animation பண்பை பயன்படுத்தி அனிமேஷனை உருவாக்கவும் கட்டுப்படுத்தவும் வேண்டும்.

Defining Keyframes

@keyframes விதி அனிமேஷன் பண்புகளின் விகிதங்களை வெவ்வேறு நிலைகளில் வரையறுக்கப் பயன்படுகிறது. இது @keyframes அல்லது @-webkit-keyframes என்ற விதியின் மூலம் வரையறுக்கப்படுகிறது. இந்த விதி சதவீதம் (%) அல்லது கீவொர்ட்கள் மூலம் வரையறுக்கப்படுகிறது, இதில் 0% என்பது தொடக்க புள்ளியையும் 100% என்பது முடிவு புள்ளியையும் குறிக்கிறது.

Important Animation Properties

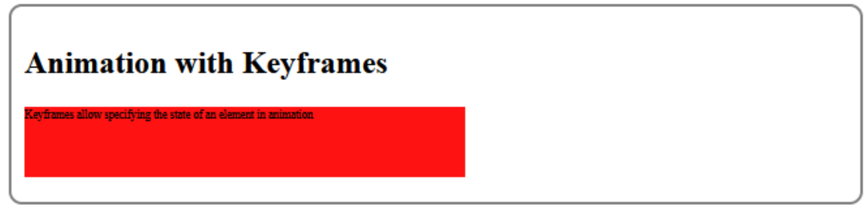
- **animation-name**: @keyframes ஐப் பயன்படுத்தி அனிமேஷனின் பெயர் வரையறுக்கப்பட்டுள்ளது.
- **animation-duration**: அனிமேஷனின் காலம் நொடிகளில்
- **animation-delay**: அனிமேஷன் தொடங்கும் முன் தாமதம்.
- **animation-iteration-count**: அனிமேஷன் எத்தனை முறை திரும்பி இயங்குகிறது என்பதை குறிக்கிறது
- **animation**: பண்பின் சுருக்கமான குறியீடாகும். இதில் பல அனிமேஷன் பண்புகளை ஒரே இடத்தில் வரையறுக்கலாம்.

Example

```
@-webkit-keyframes zoom{
   0% { width: 0px; height: 0px; font-size: 0px;}
   25% { background:RED;}
   100% { width: 100%; height: 100px; font-size: auto;}
}
@keyframes zoom{
   0% { width: 0px; height: 0px; font-size: 0px;}
   25% { background:RED;}
   100% { width: 100%; height: 100px; font-size:auto;}
}
p{
   -webkit-animation: 5s 2s zoom 2;

   animation: 5s 2s zoom 2;
}
<article>
   <h2>Animation with Keyframes</h2>
   <p>
      Keyframes allow specifying the state of an element in animation
   </p>
</article>
```

Output



Pagination

CSS pagination என்பது நீண்ட வலைப்பக்கங்களை பல பக்கங்களாகப் பிரிக்கும் ஒரு தொழில்நுட்பம் ஆகும். இது அடிக்கடி கட்டுரைகள், ப்ளாக் போஸ்டுகள், அல்லது ஒரு பக்கத்தில் சேர்க்க முடியாத உள்ளடக்கத்திற்காகப் பயன்படுத்தப்படுகிறது.

- **Divide Content:** உங்கள் உள்ளடக்கத்தை தர்க்கரீதியான பக்கங்களாகப் பிரிக்க <div> அல்லது <section> போன்ற HTML கூறுகளைப் பயன்படுத்துங்கள்.
- **Hide Pages:** முதலில், CSS ஐப் பயன்படுத்தி முதல் பக்கத்தைத் தவிர அனைத்து பக்கங்களையும் மறைக்கவும்.
- **Create Navigation:** பயனர்கள் பக்கங்களுக்கு இடையே செல்ல அனுமதிக்கும் navigation கூறுகளை (buttons, links) சேர்க்கவும்.
- **Show/Hide Pages:** Navigationனுடன் பயனரின் தொடர்புக்கு ஏற்ப பக்கங்களை காட்டு அல்லது

மறைக்க JavaScript அல்லது CSS ஐப் பயன்படுத்துங்கள்.

Example

```
.pag-cont {
        max-width: 600px;
        margin: 0 auto;
        padding: 20px;
 }

 .content-page {
        display: none;
        padding: 20px;
        border: 1px solid #ccc;
        border-radius: 10px;
 }

 .content-page.active {
        display: block;
 }

 .pagination {
        text-align: center;
        margin-top: 10px;
 }

 .pagination a {
        display: inline-block;
        padding: 5px 10px;
        margin-right: 5px;
        text-decoration: none;
        color: #000;
        border: 1px solid #ccc;
        cursor: pointer;
}

.pagination a.active {
        background-color: #333333;
        color: #ffffff;
}

.pagination a:hover:not(.active) {
        background-color: #dddddd;
        border-radius: 5px;
}
<article>
<div class="pag-cont">
        <div class="content-page " id="page1">
        <h2>Page 1</h2>
        <p>This is the content of Page 1.</p>
        </div>
        <div class="content-page active" id="page2">
        <h2>Page 2</h2>
        <p>This is the content of Page 2.</p>
        </div>
        <div class="content-page " id="page3">
        <h2>Page 3</h2>
        <p>This is the content of Page 3.</p>
        </div>
        <div class="pagination">
        <a href="#" >1</a>
        <a href="#" class="active">2</a>
```

```
        <a href="#" >3</a>
        </div>
  </div>
  </article>
```

Output

Multiple Columns

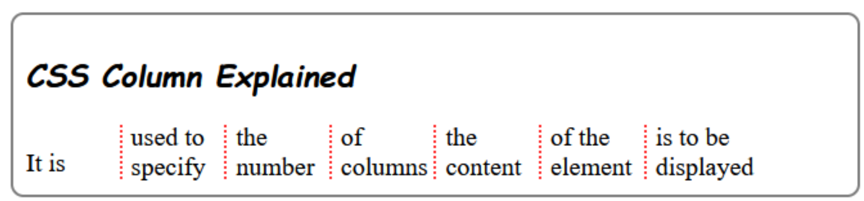
CSS பல நெடுவரிசைகள் ஒரு கூறுக்குள் உள்ள உள்ளடக்கத்தை செய்தித்தாள் அல்லது பத்திரிகை தளவமைப்பு போல இரண்டு அல்லது அதற்கு மேற்பட்ட நெடுவரிசைகளாகப் பிரிக்க அனுமதிக்கின்றன. இந்த அம்சம் உரை நிறைந்த உள்ளடக்கத்தின் படிக்கத்தக்க தன்மை மற்றும் காட்சிப் பிரசாதத்தை மேம்படுத்த மிகவும் மதிப்புமிக்கது. இதில் பின்வரும் பண்புகள் உள்ளன:

- **column-count**: உள்ளடக்கம் விநியோகிக்கப்பட வேண்டிய நெடுவரிசைகளின் எண்ணிக்கையை இது குறிப்பிடுகிறது.
- **column-width**: px இல் நெடுவரிசையின் அகலத்தை அமைக்க இது பயன்படுத்தப்படுகிறது
- **column-fill**: நெடுவரிசைகளை எவ்வாறு நிரப்புவது என்பதைக் குறிப்பிட இது பயன்படுத்தப்படுகிறது. இது மதிப்புகளின் balance மற்றும் auto எடுக்கும்.
- **column-gap**: நெடுவரிசைகளுக்கு இடையிலான இடைவெளியைக் குறிப்பிட இது பயன்படுகிறது. இது சாதாரண அல்லது பிக்சல் மதிப்பை எடுக்கும்.
- **column-rule**: நெடுவரிசைகளுக்கு இடையே உள்ள ஆட்சிக் கோட்டின் அகலம், நடை மற்றும் வண்ணத்தைக் குறிப்பிட இது பயன்படுகிறது.
- **column-span**: ஒரு உறுப்பு முழுவதும் பரவ வேண்டிய நெடுவரிசைகளைக் குறிப்பிட இது பயன்படுகிறது.

Example

```
 section{
    column-width: 60px;
    column-fill: balanced;
    column-gap: normal;
    column-rule: 2px dotted red;
 }
 section > h2{
    font-style: oblique;
    font-family: cursive;
    column-span: all;
 }
 <section>
        <h2>CSS Column Explained</h2>
        <p>
        It is used to specify the number of columns the content of the element is to be displayed
        </p>
 </section>
```

Output



User Interface

CSS பயனர் இடைமுக பண்புகள் ஒரு கூறு இடைமுக பண்பை மாற்ற அனுமதிக்கின்றன. CSS இல் உள்ள resize பண்பு ஒரு கூறு மறுஅளவிடக்கூடியதாக இருக்கும் என்பதை குறிப்பிடப் பயன்படுகிறது. இது செங்குத்து, கிடைமட்ட, இரண்டும் அல்லது இல்லை என்ற மதிப்புகளை எடுக்கும்.

Syntax

resize: vertical;

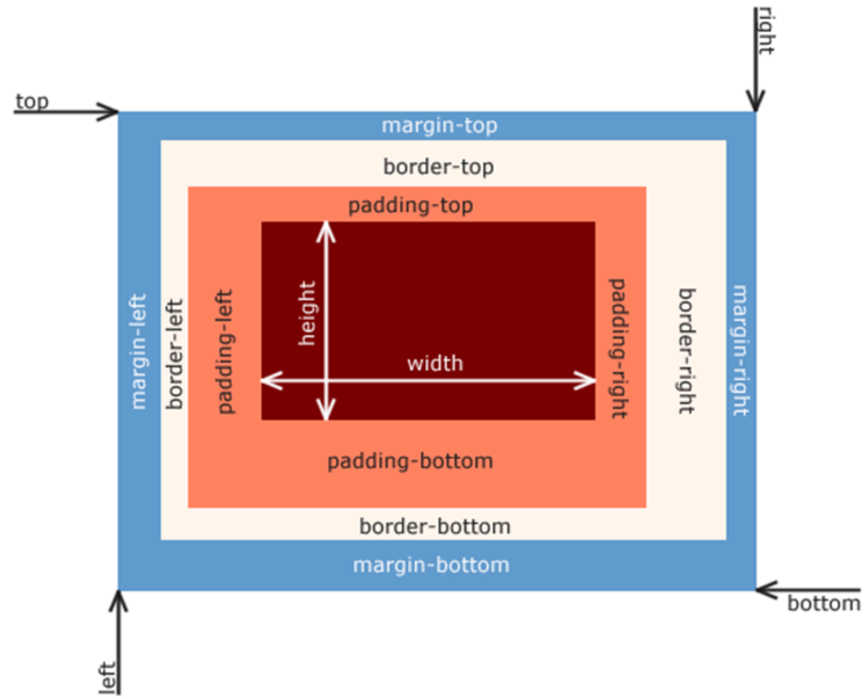
CSS filters

Filter property ஒன்று அல்லது அதற்கு மேற்பட்ட வடிகட்டி செயல்பாடுகளை ஏற்றுக்கொள்கிறது, ஒவ்வொன்றும் படங்களில் வெவ்வேறு காட்சி விளைவைக் குறிப்பிடுகிறது. Filter propertyகளைப் பயன்படுத்தி இது பயன்படுத்தப்படுகிறது. சில filterகள்

- blur(px);
- grayscale(%);
- brightness(%);
- opacity(%);
- sepia(%);

Box Model

CSS box model வலை வடிவமைப்பு மற்றும் தளவமைப்பில் ஒரு அடிப்படை கருத்தாகும். இது ஒரு வலைப் பக்கத்தில் உள்ள கூறுகள் செவ்வக பெட்டிகளில் எவ்வாறு வரையப்படுகின்றன என்பதை விவரிக்கிறது, ஒவ்வொரு பெட்டியும் content, padding, border மற்றும் margin உள்ளிட்ட பல அடுக்குகளைக் கொண்டுள்ளது.



Positioning

CSS (Cascading Style Sheets) positioning என்பது ஒரு அடிப்படைக் கருத்தாகும், இது வலை டெவலப்பர்களை வலைப்பக்கத்தில் உள்ள HTML உறுப்புகளின் தளவமைப்பு மற்றும் இடங்களை கட்டுப்படுத்த அனுமதிக்கிறது. CSS positioningன் வகைகள்

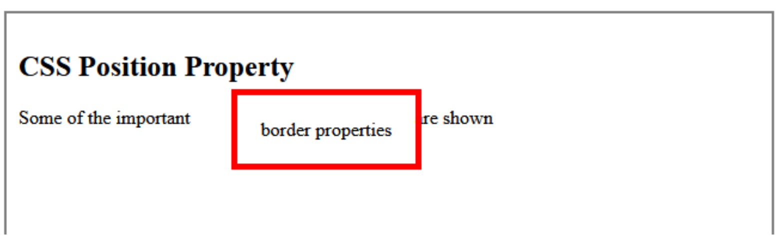
- **static**: ஆவண ஓட்டத்தின் படி கூறுகள் அவற்றின் இயல்பு நிலையில் காட்டப்படும்.
- **relative**: ஆவண ஓட்டத்தில் அவற்றின் இயல்புநிலை நிலைக்கு தொடர்புடைய உறுப்புகள் நிலைநிறுத்தப்படுகின்றன.
- **fixed**: Viewport (the browser window) தொடர்பான கூறுகள் நிலைநிறுத்தப்பட்டுள்ளன. பயனர் பக்கத்தை ஸ்க்ரோல் செய்யும் போதும் அவை அவற்றின் நிலையிலேயே நிலைத்திருக்கும்.

- **absolute**: உறுப்புகள் நிலையற்ற நிலையில் (relative, absolute, or fixed) அருகிலுள்ள மூதாதையருடன் தொடர்புடையதாக இருக்கும்.
- **sticky**: உறுப்புகள் பயனர் ஸ்க்ரோலின் அடிப்படையில் அமைந்திருக்கும்.

<u>Example</u>

```
span{
   position: relative;
   top: 10px;
   left: 20px;
   border: 5px solid RED;
   margin: 10px;
   padding: 20px;
}
<article>
   <h2>CSS Position Property</h2>
   <p>
      Some of the important <span>border properties</span>
are shown
   </p>
</article>
```

<u>Output</u>



Viewport

வியூபோர்ட் என்பது தற்போது ஒரு சாதனத்தின் திரையில் காட்டப்படும் ஒரு வலைப் பக்கத்தின் தெரியும் பகுதியாகும். இது உங்கள் வலைப் பக்கம் வரையப்படும் செவ்வக பகுதியாகும், இது பொதுவாக சாதனத்தின் திரையின் அளவுக்கு சமமாக இருக்கும். வியூபோர்டின் நடத்தையை கட்டுப்படுத்த, வியூபோர்ட் மெட்டா டேக் HTML ஆவணத்தின் <head> பிரிவில் சேர்க்கப்படுகிறது.

<u>Syntax</u>

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

Flex Box

CSS Flexbox, Flexible Box தளவமைப்பு என்றும் அறியப்படுகிறது, சிக்கலான மற்றும் பதிலளிக்கும் தளவமைப்புகளை உருவாக்க வடிவமைக்கப்பட்ட CSS இல் உள்ள ஒரு தளவமைப்பு மாதிரியாகும். இது கூறுகளின் அளவு தெரியாத அல்லது இயக்கவியலாக இருக்கும்போது கூட ஒரு கொள்கலன் உள்ளே இடத்தை விநியோகித்து உள்ளடக்கத்தை சீரமைக்க அனுமதிக்கிறது. இது பின்வருவனவற்றை குறிப்பிடப் பயன்படுகிறது

- Flex Container
- Flex Items

Flex Container

இந்த உருப்படியில் ஃப்ளெக்ஸ் பொருட்களின் தொகுப்பை கொண்டுள்ள உருப்படி இதுவும் "ஃப்ளெக்ஸ் கண்டைனர்" என்றும் அழைக்கப்படுகிறது. இதை நிறைவேற்றுவதற்கு display பண்புடன் flex அல்லது inline-flex என்ற மதிப்பை பயன்படுத்துகிறீர்கள். இந்த கண்டைனருக்கு அதிக பண்புகள் உள்ளன.

- flex-direction: [column, row/-reverse]

- flex-wrap: [wrap/nowrap]
- justify-content: [center, flex-start, flex-end, space-between, space-around]

Flex Items

நெகிழ்வான கொள்கலன் உள்ளே உள்ள தனிப்பட்ட கூறுகள் கொள்கலனின் பண்புகளின் அடிப்படையில் சீரமைக்கப்பட்டு விநியோகிக்கப்படுகின்றன. இது பின்வரும் பண்புகளை எடுக்கும்

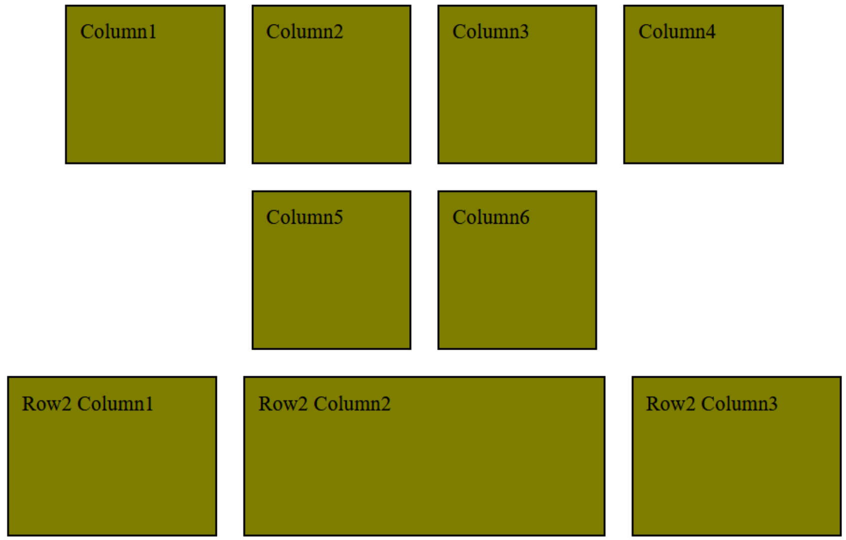
- **flex-grow**: இந்த சொத்து மீதமுள்ள உறுப்புடன் தொடர்புடைய ஒரு உறுப்பை வளர்க்கிறது.
- **flex-shrink**: இந்த பண்பு மீதமுள்ள உறுப்புடன் தொடர்புடைய ஒரு உறுப்பை சுருக்குகிறது.

Example

```
.container{
   display: flex;
   flex-direction: row;
   flex-wrap: wrap;
   justify-content: center;

}
.col{
   width: 100px;
   height: 100px;
   border: 2px solid black;
   background-color: olive;
   padding: 10px;
   margin: 10px;
}
.lg-1{
   flex-grow: 1;
}
.lg-4{
   flex-grow: 4;
}
<div class="container">
   <div class="col">Column1</div>
   <div class="col">Column2</div>
   <div class="col">Column3</div>
   <div class="col">Column4</div>
   <div class="col">Column5</div>
   <div class="col">Column6</div>
</div>
<div class="container">
   <div class="col lg-1">Row2 Column1</div>
   <div class="col lg-4">Row2 Column2</div>
   <div class="col lg-1">Row2 Column3</div>
</div>
```

Output



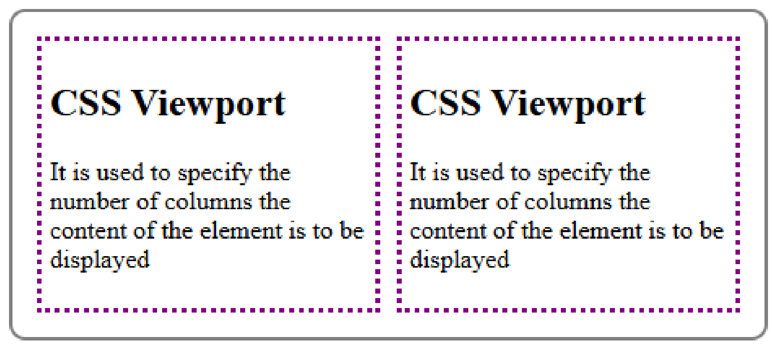
Fluid Width

CSS இல், நெகிழ் அகலம் என்பது ஒரு கூறு அதன் கொள்கலன் உள்ள இடத்தைப் பொறுத்து அதன் அகலத்தை இயக்கவியல் ரீதியாக சரிசெய்யும் திறனை குறிக்கிறது. இது வெவ்வேறு திரை அளவுகள் மற்றும் சாதனங்களுக்கு ஏற்ப மாற்றியமைக்கும் பதிலளிக்கும் வடிவமைப்புகளை உருவாக்க ஏற்றது. இதை செயல்படுத்த ஒரு கூறின் அகல பண்பு px மதிப்பிற்கு பதிலாக % இல் அமைக்கப்படுகிறது.

Example

```
  div{
    width: 50%;
    padding: 5px;
    margin: 5px;
    border: 3px dotted purple;
  }
<article>
  <div>
   <h2>CSS Viewport</h2>
   <p>
     It is used to specify the number of columns the content of the element is to be displayed
   </p>
  </div>
  <div>
   <h2>CSS Viewport</h2>
   <p>
     It is used to specify the number of columns the content of the element is to be displayed
   </p>
  </div>
</article>
```

Output



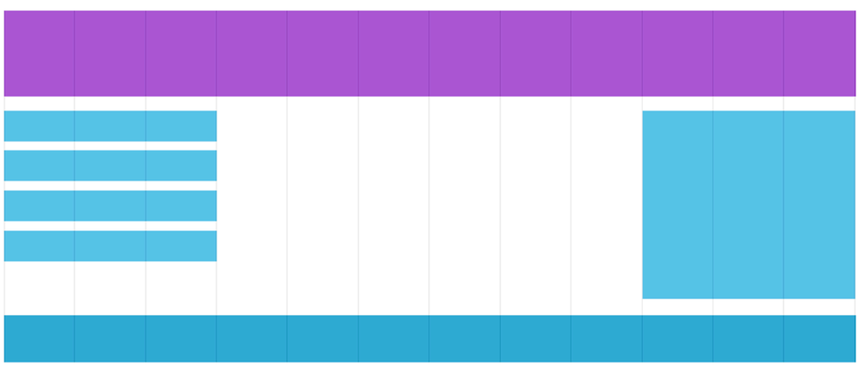
Grids

CSS ஐப் பயன்படுத்தி சிறந்த வடிவமைப்புக்காக ஒரு வலைப் பக்கம் கிரிடுகளை அடிப்படையாகக் கொண்டு கட்டப்படுகிறது. இதில் ஒரு வலைப் பக்கம் முழு 100% அகலத்திற்கும் 12 நெடுவரிசைகளாகப் பிரிக்கப்படுகிறது.

நெகிழ் அகல பண்பு ஒரு வலைப் பக்கத்தில் நெடுவரிசைகளை பிரித்து இணைக்கப் பயன்படுகிறது. 100% அகலம் 12 நெடுவரிசைகளால் பிரிக்கப்பட்டு ஒரு நெடுவரிசையின் அகலத்தை 8.33% ஆகக் கிடைக்கிறது. ஒரு வலைப் பக்கம் மேற்கண்ட நெடுவரிசை தளவமைப்பைப் பயன்படுத்தி வடிவமைக்கப்பட்டுள்ளது, அதன் பொது வடிவம் கீழே கொடுக்கப்பட்டுள்ளது.

Example

```
.col-1{
   width: 8.33%;/*Single column width*/
 }
.col-2{
  width: 16.66%;/*Double column width*/
}
```



Media Queries

Media queriesகள் CSS இல் ஒரு சக்திவாய்ந்த கருவியாகும், இது வெவ்வேறு திரை அளவுகள், திசைகள் மற்றும் சாதன திறன்களுக்கு ஏற்ப மாற்றியமைக்கும் பதிலளிக்கும் வலை வடிவமைப்புகளை உருவாக்க உங்களை அனுமதிக்கிறது. அவை சாதனத்தின் பண்புகள் அல்லது பயனரின் பார்வை சூழலைப் பொறுத்து குறிப்பிட்ட பாணிகளைப் பயன்படுத்த உங்களை அனுமதிக்கின்றன.

Syntax

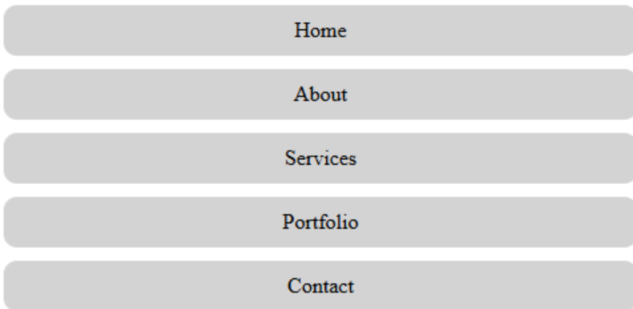
```
@media mediaType and (mediaFeature) {
        /* CSS rules to apply when the conditions are met */
}
```

- **mediaType**: இது வலைப்பக்கம் காட்டப்படும் ஊடகத்தைக் குறிப்பிடுகிறது. எடுத்துக்காட்டு திரை, அச்சு, பேச்சு
- **mediaFeature**: கொடுக்கப்பட்ட cssஐப் பயன்படுத்துவதற்கு சரிபார்க்க வேண்டிய நிபந்தனையை இது வரையறுக்கிறது. எடுத்துக்காட்டு அகலம், உயரம், அதிகபட்ச அகலம், நிமிட அகலம், நோக்குநிலை

Example

```
/* Media query for screens smaller than 600px */
@media screen and (max-width: 600px) {
  .menu {
     flex-direction: column; /* Change to a vertical layout */
    text-align: center; /* Center-align the items */
  }
 .menu li {
    display: block; /* Each item on a new line */
    margin-bottom: 10px; /* Add some spacing between items */
```

```
 }
a{
  width: 100%;
 }
```




```
}
```

<u>Output</u>