

# CSS



# Apa yang Kita Pelajari?

**Memulai CSS**

**Selector**

**Color**

**Background**

**Font & Text**

**Link**

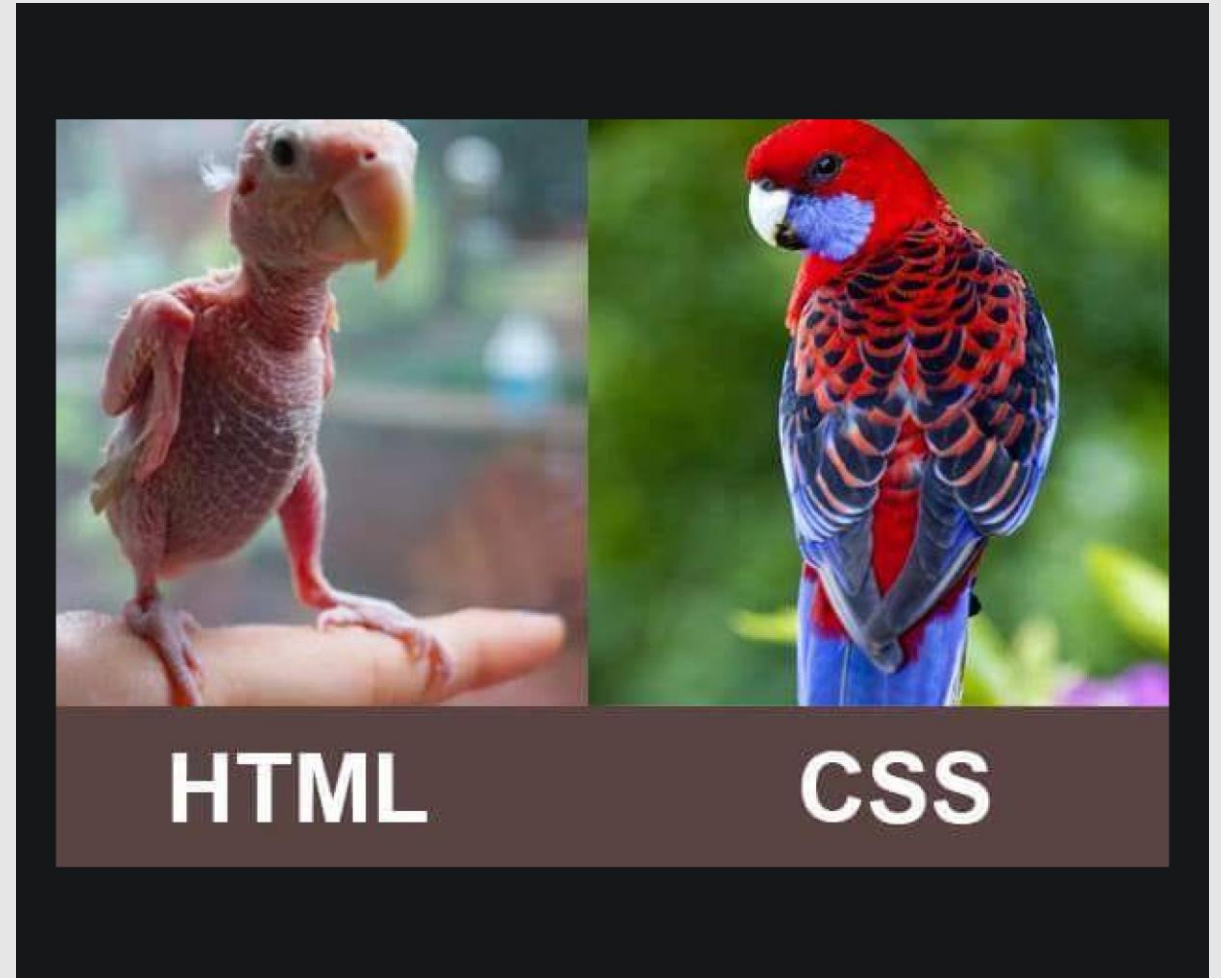
**List**

**Table**

# Memulai CSS

# Apa itu CSS

- CSS (Cascading Style Sheets) adalah bahasa pemrograman yang digunakan untuk mengatur tampilan dan gaya dari halaman web.
- Analogi sederhana untuk CSS adalah seperti gambar disamping.



# Style CSS

- CSS (Cascading Style Sheets) diperkenalkan pada bulan Desember 1996 oleh World Wide Web Consortium (W3C) untuk memberikan cara yang lebih baik untuk menata halaman web.
- Style CSS bisa ditambahkan atau diterapkan ke halaman HTML dengan 3 cara:
  - **Inline:** menggunakan atribut style di tag awal HTML.
  - **Embedded:** menggunakan elemen `<style>` di bagian `<head>`.
  - **External:** menggunakan elemen `<link>` yang merujuk ke file CSS eksternal.

# Inline Style

- Digunakan untuk menerapkan aturan style unik ke suatu elemen dengan menempatkan aturan CSS langsung ke tag awal.

```
<h1 style="color:red; font-size:30px;">  
  This is a heading  
</h1>  
<p style="color:green; font-size:18px;">  
  This is a paragraph.  
</p>  
<div style="color:green; font-size:18px;">  
  This is some text.  
</div>
```

# Embedded Style

- Embedded atau internal style hanya mempengaruhi dokumen tempat CSS disematkan.
- Embedded style didefinisikan di bagian <head> dengan menggunakan <style>.

```
<head>
  <style>
    body { background-color: YellowGreen; }
    h1 { color: blue; }
    p { color: red; }
  </style>
</head>
```

# External Style

- External style dapat ditautkan ke dokumen HTML menggunakan tag `<link>`.
- Tag `<link>` berada di dalam bagian `<head>`.

```
<head>  
  <link rel="stylesheet" href="css/style.css">  
</head>
```



# Menggunakan @import

- Statement @import menginstruksikan browser untuk memuat CSS eksternal di dalam file CSS.

```
<style>  
  @import url("css/style.css");  
  p {  
    color: blue;  
    font-size: 16px;  
  }  
</style>
```

// atau

```
@import url("css/layout.css");  
@import url("css/color.css");  
body {  
  color: blue;  
  font-size: 14px;  
}
```

# Selector

# Selector

- Selector adalah pola untuk mencocokkan elemen pada halaman web.
- Selector digunakan untuk menargetkan elemen tertentu pada halaman web agar dapat diberi style.

# Universal Selector

- Selector universal dilambangkan dengan tanda bintang (\*) dan akan style-nya diterapkan ke setiap elemen dalam dokumen HTML.
- Selector ini sering digunakan untuk menghapus margin dan padding default dari elemen (reset css).

```
* {  
  margin: 0;  
  padding: 0;  
}
```

# Element Selector

- Element selector digunakan untuk memilih elemen HTML berdasarkan nama elemennya.
- Misalnya, untuk memilih semua paragraf pada halaman web, kita dapat menggunakan selector **p**.

```
p {  
    color: blue;  
}
```

# ID Selector

- ID selector digunakan untuk memilih elemen HTML berdasarkan id-nya.
- ID selector didefinisikan dengan tanda hash (#) diikuti dengan nilai id-nya.

```
#text {  
  color: blue;  
}
```

# Class Selector

- Class selector digunakan untuk memilih elemen HTML berdasarkan class-nya.
- Class selector didefinisikan dengan tanda titik diikuti dengan nilai class-nya.

```
.text {  
  color: blue;  
  background: #ffaaff;  
}
```

# Class Selector

- Class selector bisa dikombinasikan dengan nama elemennya untuk memberikan style khusus.
- **p.text** artinya kita hanya memberikan style berdasarkan class text namun untuk elemen paragraf saja.

```
p.text {  
  color: blue;  
  font-size: 18px;  
  font-weight: bold;  
}
```



# Descendant Selector

- Descendant Selector digunakan untuk memilih elemen anak (descendant) yang berada dalam elemen induk (parent).

```
h1 em {  
  color: green;  
}  
  
ul.menu {  
  padding: 0;  
  list-style: none;  
}  
  
ul.menu li {  
  display: inline;  
}  
  
ul.menu li a {  
  margin: 10px;  
  text-decoration: none;  
}
```

```
<h1>This is a <em>heading</em>  
</h1>  
<ul class="menu">  
  <li>  
    <a href="#">Home</a>  
  </li>  
  <li>  
    <a href="#">About</a>  
  </li>  
  <li>  
    <a href="#">Services</a>  
  </li>  
  <li>  
    <a href="#">Contact</a>  
  </li>  
</ul>
```

This is a *heading*

[Home](#) [About](#) [Services](#) [Contact](#)

# Child Selector

- Child selector digunakan untuk memilih hanya elemen yang merupakan anak langsung dari elemen induknya.
- Child selector disimbolkan dengan tanda lebih besar dari (>).

```
<ul>
  <li>
    <a href="#">Home</a>
  </li>
  <li>
    <a href="#">About</a>
  </li>
  <li>
    <a href="#">Services</a>
    <ol>
      <li>
        <a href="#">Design</a>
      </li>
      <li>
        <a href="#">Development</a>
      </li>
    </ol>
  </li>
  <li>
    <a href="#">Contact</a>
  </li>
</ul>
```

```
<style>
  ul > li {
    list-style: square;
  }
  ul > li ol {
    list-style: none;
  }
</style>
```

# Adjacent Sibling Selector

- Adjacent sibling selector dapat digunakan untuk memilih elemen saudara (yaitu elemen pada tingkat yang sama).
- Pada contoh ini, Adjacent Sibling Selector digunakan untuk memilih elemen **p** yang langsung berada setelah elemen **h1**, serta memilih elemen **span** yang langsung berada setelah elemen **strong**.

```
<style>
/* Memilih elemen <p> yang langsung berada setelah elemen <h1>
*/
h1 + p {
  color: blue;
  font-size: 20px;
}

/* Memilih elemen <span> yang langsung berada setelah elemen
<strong> */
strong + span {
  text-decoration: underline;
}
</style>

<body>
  <h1>Judul</h1>
  <p>Paragraf pertama yang berada setelah judul.</p>
  <span>Teks dalam elemen span.</span>
  <strong>Teks tebal</strong>
  <span>Teks dalam elemen span setelah teks tebal.</span>
</body>
```

## Judul

Paragraf pertama yang berada setelah judul.

Teks dalam elemen span. **Teks tebal** Teks dalam elemen span setelah teks tebal.

# General Sibling Selector

- General Sibling Selector memungkinkan Anda memilih semua elemen saudara yang ada setelah elemen referensi dengan hubungan saudara, tanpa memerlukan hubungan yang langsung seperti pada Adjacent Sibling Selector.
- Pada contoh di ini, General Sibling Selector digunakan untuk memilih semua elemen **p** yang berada setelah elemen **h1**, serta memilih semua elemen **span** yang berada setelah elemen **strong**.

```
<style>
  h1 ~ p {
    color: blue;
    font-size: 20px;
  }
```

```
  strong ~ span {
    text-decoration: underline;
  }
```

```
</style>
```

```
<body>
```

```
  <h1>Judul</h1>
```

```
  <p>Paragraf pertama yang berada setelah judul.</p>
```

```
  <p>Paragraf kedua yang juga berada setelah judul.</p>
```

```
  <span>Teks dalam elemen span.</span>
```

```
  <strong>Teks tebal</strong>
```

```
  <span>Teks dalam elemen span setelah teks tebal.</span>
```

```
</body>
```

## Judul

Paragraf pertama yang berada setelah judul.

Paragraf kedua yang juga berada setelah judul.

Teks dalam elemen span. **Teks tebal** Teks dalam elemen span setelah teks tebal.

# Grouping Selector

- Grouping selector berguna ketika kita ingin menerapkan style yang sama untuk beberapa elemen yang berbeda.

```
<style>
```

```
h1, h2 {  
  color: blue;  
  font-family: "Arial", sans-serif;  
}
```

```
.teks, .keterangan {  
  font-size: 16px;  
  line-height: 1.5;  
}
```

```
</style>
```

```
<body>
```

```
<h1>Judul Utama</h1>
```

```
<p class="teks">Ini adalah paragraf dengan teks biasa.</p>
```

```
<h2>Judul Kedua</h2>
```

```
<p class="keterangan">
```

```
  Ini adalah paragraf dengan keterangan.
```

```
</p>
```

```
</body>
```

## Judul Utama

Ini adalah paragraf dengan teks biasa.

## Judul Kedua

Ini adalah paragraf dengan keterangan.

# Color

# Color

- Properti color digunakan untuk menentukan warna teks dari suatu elemen.
- Warna dalam CSS paling sering ditentukan dalam format:
  - Keyword color seperti "red", "green", "blue", "transparent" dan lain-lain.
  - Nilai hex seperti "#ff0000", "#00ff00" dan lain-lain.
  - Nilai RGB seperti "rgb(255, 0, 0)".

```
/* menggunakan keyword */  
  
h1 {  
  color: red;  
}  
  
/* menggunakan hex */  
  
h2 {  
  color: #ffa500;  
}  
  
/* menggunakan rgb */  
  
h3 {  
  color: rgb(255, 165, 0);  
}
```

# Background



# Background

- Background memainkan peran penting dalam presentasi visual halaman web.
- CSS menyediakan beberapa properti untuk menata background suatu elemen, termasuk memberikan warna, menempatkan gambar, mengatur posisi gambar di background dan lain-lain.
- Properti background terdiri dari **background-color**, **background-image**, **background-repeat**, **background-attachment** dan **background-position**.

# Background Color

- Properti background-color digunakan untuk mengatur warna latar belakang suatu elemen.
- Format background color bias berbentuk hex, keyword warna dan rgb.

```
body {  
    background-color: #f0e68c;  
}
```

# Background Image

- Properti **background-image** menggunakan gambar sebagai latar belakang elemen HTML.
- Secara default, browser akan menyusun (repeat) gambar latar belakang baik secara horizontal maupun vertikal untuk mengisi seluruh area elemen.

```
body {  
    background-image: url("images/tile.png");  
}
```

# Background Repeat

- Kita dapat menyetel gambar latar belakang untuk diulang secara vertikal (sumbu y), horizontal (sumbu x) dengan property **background-repeat**.

```
body {  
  background-image: url("images/gradient.png");  
  background-repeat: repeat-x;  
}
```

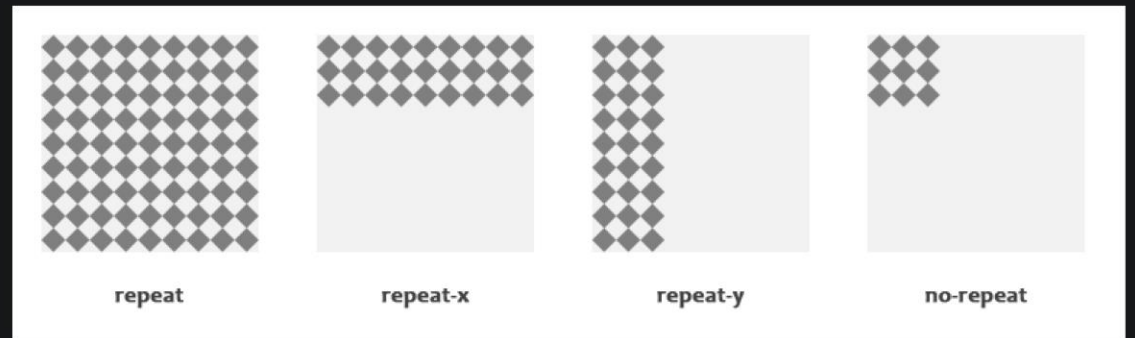
# Background Repeat

- Kita juga dapat menggunakan nilai **repeat-y** untuk mengulang gambar latar belakang secara vertikal sepanjang sumbu y, atau nilai **no-repeat** untuk mencegah pengulangan sama sekali.

```
body {  
  background-image: url("images/texture.png");  
  background-repeat: no-repeat;  
}  
  
div.header {  
  background-image: url("images/texture.png");  
  background-repeat: repeat-y;  
}
```

# Background Repeat

- Ilustrasi bentuk repeat dari background.



# Font & Text

# Font

- CSS memiliki beberapa properti yang digunakan untuk mengatur style dan tampilan font dan teks pada elemen HTML.



# font-family

- Properti font-family digunakan untuk mengatur jenis font atau style huruf yang akan digunakan pada teks.
- Kita dapat menyediakan daftar font dengan prioritas yang berbeda jika font pertama tidak tersedia pada perangkat pengguna.

```
p {  
  font-family: "Arial", sans-serif;  
}
```

# font-size

- Properti font-size digunakan untuk mengatur ukuran font teks.
- Ukuran font dapat ditentukan dalam berbagai unit seperti piksel (px), em, rem, persentase (%), dan lainnya.

```
h1 {  
  font-size: 32px;  
}  
  
p {  
  font-size: 16px;  
}
```

# font-weight

- Properti font-weight digunakan untuk mengatur ketebalan font teks.
- Nilai yang umum digunakan adalah "normal" (400) dan "bold" (700), tetapi ada juga nilai angka dari 100 hingga 900 untuk ketebalan yang lebih spesifik.

```
h1 {  
    font-weight: bold;  
}  
  
p {  
    font-weight: normal;  
}
```

# font-style

- Properti font-style digunakan untuk mengatur style teks seperti miring (italic) atau normal.

```
em {  
    font-style: italic;  
}  
  
span {  
    font-style: normal;  
}
```

# text-decoration

- Properti text-decoration digunakan untuk mengatur style dekorasi teks seperti garis bawah (underline), garis tengah (line-through) dan lainnya.

```
a {  
  text-decoration: none;  
  /* Menghapus dekorasi  
    link default */  
}  
  
p {  
  text-decoration: underline;  
}
```

# text-align

- Properti text-align digunakan untuk mengatur penataan teks dalam sebuah elemen, seperti "left" (kiri), "center" (tengah), "right" (kanan) atau "justify" (rata kiri dan kanan).

```
p {  
  text-align: center;  
}
```

# Link

# Menata Link

- Ada empat status berbeda dari sebuah tautan (link), yaitu link, visited, active dan hover.
- **Link:** adalah status normal dari link sebelum pengguna mengklik atau mengunjungi tautan tersebut.
- **Visited:** adalah status tautan setelah pengguna mengklik dan mengunjungi halaman yang dituju oleh link tersebut.
- **Active:** adalah status tautan ketika pengguna sedang menekan (klik dan tahan) tautan.
- **Hover:** adalah status tautan ketika kursor pengguna berada di atas tautan.

```
a:link {  
    /* unvisited link */  
    color: #ff0000;  
    text-decoration: none;  
    border-bottom: 1px solid;  
}  
a:visited {  
    /* visited link */  
    color: #ff00ff;  
}  
a:hover {  
    /* mouse over link */  
    color: #00ff00;  
    border-bottom: none;  
}  
a:active {  
    /* active link */  
    color: #00ffff;  
}
```

[Visit React](#)



# Membuat Tombol dari Link

- Kita dapat mengubah tampilan link teks biasa agar terlihat seperti tombol menggunakan CSS.

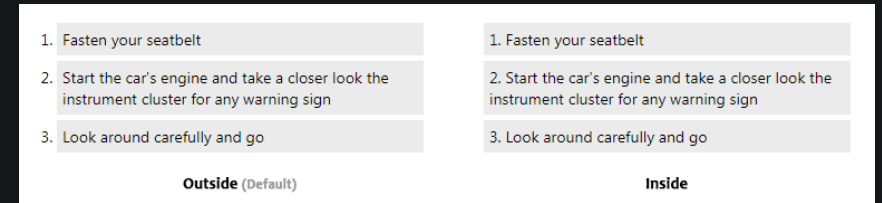
CSS Link Button

```
a:link, a:visited {  
    color: white;  
    background-color: #1ebba3;  
    display: inline-block;  
    padding: 10px 20px;  
    border: 2px solid #099983;  
    text-decoration: none;  
    text-align: center;  
    font: 14px Arial, sans-serif;  
}  
  
a:hover, a:active {  
    background-color: #9c6ae1;  
    border-color: #7443b6;  
}
```

# List

# Mengubah Posisi Marker List

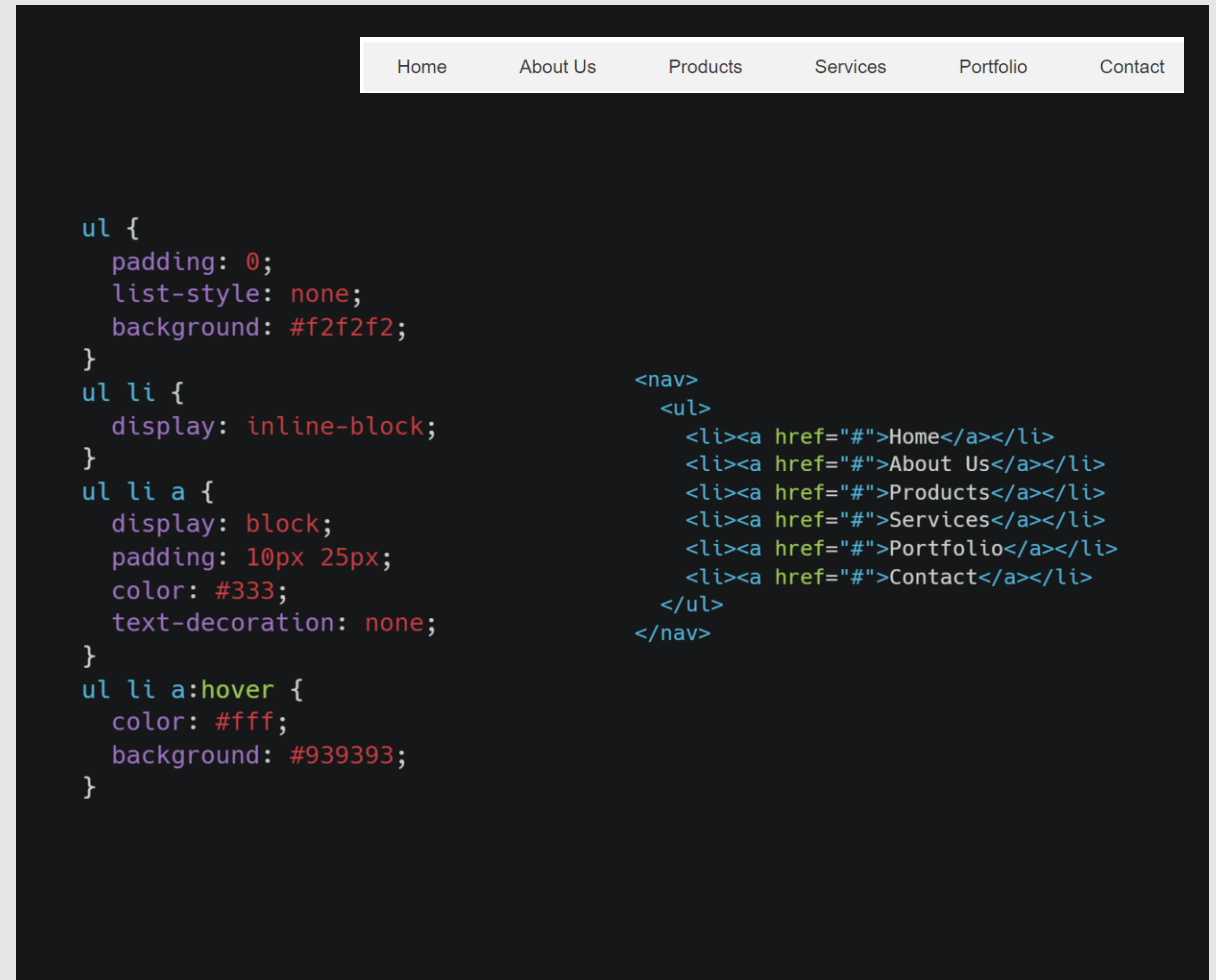
- Secara default, marker (penanda) setiap item list diposisikan di luar.
- Kita dapat memposisikan marker atau poin-poin item menggunakan properti **list-style-position**.



```
ol.in li {  
    list-style-position: inside;  
}  
  
ol.out li {  
    list-style-position: outside;  
}
```

# Membuat Menu Navigasi

- List HTML sering digunakan untuk membuat menu navigasi horizontal atau menu yang biasanya muncul di bagian atas situs web.



# Table

# Alignment pada Sel

- Untuk meratakan teks secara horizontal di dalam sel tabel, Kita dapat menggunakan properti text-align.
- Kita juga dapat meratakan teks secara vertikal di dalam elemen <th> dan <td> ke atas, bawah atau tengah menggunakan properti vertical-align.

```
th {  
    text-align: left;  
}  
  
/* atau */  
  
th {  
    height: 40px;  
    vertical-align: bottom;  
}
```

# Membuat Table Zebra Strip

- Menetapkan warna latar belakang yang berbeda untuk baris ganjil dan genap adalah teknik untuk meningkatkan keterbacaan tabel yang memiliki data dalam jumlah besar.

Row	First Name	Last Name	Email
1	Clark	Kent	clarkkent@mail.com
2	John	Carter	johncarter@mail.com
3	Peter	Parker	peterparker@mail.com

```
table {  
  width: 100%;  
  font-family: arial, sans-serif;  
  border-collapse: collapse;  
}  
th, td {  
  padding: 8px;  
  text-align: left;  
  border-top: 1px solid #dee2e6;  
}  
tbody tr:nth-child(odd) {  
  background-color: #f2f2f2;  
}
```

# Membuat Table Responsive

- Agar table dapat ditampilkan dengan baik di mobile, kita dapat menambahkan responsive ke table dengan mengaktifkan scroll horizontal.

```
<div style="overflow-x: auto;">  
  <table>  
    ... table content ...  
  </table>  
</div>
```

ID	Supplier	Contact
1	Exotic Liquids	Charlott
2	New Orleans Cajun Delights	Shelley
3	Grandma Kellys Homestead	Regina
4	Tokyo Traders	Yoshi
5	Mayumis	Mayumi