

SVG (Scalable Vector Graphic)

SVG adalah format gambar berbasis XML yang digunakan untuk membuat gambar berkualitas tinggi dan dinamis. SVG menggunakan bentuk dasar seperti garis, lingkaran, dan poligon untuk membuat gambar, berbeda dari format gambar raster seperti JPEG, PNG, atau GIF yang menggunakan grid piksel.

- SVG bersifat skalar, artinya mereka mempertahankan kualitas pada ukuran berapa pun tanpa meningkatkan ukuran file.

- XML membuatnya mudah dibaca oleh manusia dan kompatibel dengan HTML.

Umumnya digunakan untuk ikon, grafik, diagram, gambar sederhana, dan pattern background. Mereka dapat dimodifikasi secara programatik melalui CSS dan JavaScript.

A. Menyisipkan SVG

Dapat dilakukan dengan 2 cara, yaitu menautkan atau menyisipkan:

1) Menautkan

Menautkan SVG berarti menggunakan file SVG eksternal dalam halaman HTML Anda. Ada dua cara utama untuk menautkan SVG:

1. Menggunakan Elemen ``:

html

Salin kode

```

```

- **Kelebihan:** Mudah digunakan dan membuat halaman lebih mudah dibaca.
- **Kekurangan:** Konten SVG tidak bisa dimanipulasi dengan CSS atau JavaScript.

2. Menggunakan CSS sebagai Latar Belakang:

css

Salin kode

```
.example {  
  background-image: url('path/to/your-image.svg');  
}
```

html

Salin kode

```
<div class="example"></div>
```

- **Kelebihan:** Berguna untuk penggunaan sebagai latar belakang elemen.
- **Kekurangan:** Sama seperti ``, konten SVG tidak bisa dimanipulasi.

2) Menyisipkan

Menyisipkan SVG berarti menambahkan kode SVG langsung ke dalam dokumen HTML. Ini memungkinkan SVG menjadi bagian dari DOM, sehingga dapat dimanipulasi dengan CSS atau JavaScript.

1. Menyisipkan Langsung dalam HTML:

html

Salin kode

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100">
  <rect x="0" y="0" width="100" height="50" />
  <circle cx="50" cy="50" r="10" />
</svg>
```

- **Kelebihan:** SVG menjadi bagian dari DOM sehingga dapat diubah dengan CSS dan JavaScript.
- **Kekurangan:** Dapat membuat kode HTML lebih panjang dan sulit dibaca, serta kurang cacheable.

B. Membuat dan Mengedit SVG

SVG Biasanya dibuat menggunakan Adobe Illustrator atau Figma. Memahami struktur SVG membantu dalam melakukan penyesuaian kecil atau kustomisasi.

[xmlns] untuk memastikan bahwa elemen-elemen SVG dikenali sebagai bagian dari spesifikasi SVG oleh browser.

```
<svg xmlns="http://www.w3.org/2000/svg" width="100" height="100">
  <circle cx="50" cy="50" r="40" />
</svg>
```

Pada contoh di atas, `xmlns` mendefinisikan bahwa semua elemen di dalam tag `<svg>` adalah bagian dari spesifikasi SVG.

[viewBox] mendefinisikan batas-batas dari SVG.

Format: `viewBox="min-x min-y width height"`

- `min-x` dan `min-y`: Koordinat dari titik asal (0,0) di sudut kiri atas dari area tampilan.
- `width` dan `height`: Lebar dan tinggi dari area koordinat yang akan ditampilkan.

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100" width="200" height="200">
  <rect x="10" y="10" width="80" height="80" fill="blue" />
</svg>
```

Pada contoh di atas:

- `viewBox="0 0 100 100"` menetapkan bahwa koordinat dari 0 ke 100 untuk kedua sumbu x dan y harus dipetakan ke dalam kotak tampilan SVG.
- `width="200" height="200"` menunjukkan ukuran tampilan dari elemen SVG di halaman web. SVG akan diskalakan untuk mengisi area ini dengan mempertahankan rasio aspek.

[fill] memberikan warna pada isi bentuk
[stroke] memberikan warna pada garis tepi bentuk

Sebagian besar gambar dibuat hanya dengan selusin atau lebih elemen-elemen dasar ini:

<rect> untuk menggambar rectangle
<circle> untuk menggambar lingkaran
<text> untuk menggambar text
<path> untuk menggambar path/garis-garis

Detail: w3school svg tutorial

C. Additional Resources

1. SVG icon libraries (free)
 - <https://fonts.google.com/icons> (Material Icons)
 - <https://feathericons.com/>
2. Make/edit your own SVG
 - <https://yqnn.github.io/svg-path-editor/>