

SVG

TUJUAN

Mahasiswa mampu mengetahui dasar – dasar pemrograman SVG melalui Notepad

PENDAHULUAN

SVG merupakan kependekan dari Scalable Vector Graphic yang digunakan untuk mendefinisikan grafis untuk web berbasis vector. SVG ini mendefinisikan grafis dalam format XML. Sesuai namanya, grafis SVG tidak akan kehilangan kualitas gambar saat diperbesar/zoom.

LANGKAH KERJA

1. Membuka Notepad
2. Menuliskan elemen svg
3. Menyimpan pada folder yang dikehendaki, dengan ekstensi *.svg
4. Membuka file svg tersebut dengan browser seperti Internet explorer, mozilla firefox, dll

SVG Bentuk Dasar

1. Bentuk Kotak/Segiempat

Elemen <rect> digunakan untuk mendefinisikan bentuk kotak. Contoh kode :

```
<svg>
<rect x="50" y="50" width="300" height="100" stroke="green" stroke-width="3" fill="none"/>
</svg>
```

Dimana :

- atribut width dan height mendefinisikan lebar dan tinggi atribut.
- x dan y mendefinisikan koordinat awal (x,y) dari kotak tersebut.
- Atribut stroke-width mendefinisikan ketebalan garis kotak (border).
- Atribut stroke mendefinisikan warna garis kotak tersebut.
- Fill mendefinisikan warna kotak tersebut.

Hasilnya :



2. Bentuk Lingkaran

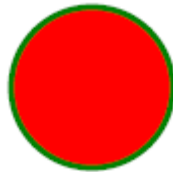
Elemen `<circle>` digunakan untuk mendefinisikan bentuk lingkaran. Contoh kode :

```
<svg>
<circle cx="100" cy="50" r="40" stroke="green" stroke-width="3" fill="red"/>
</svg>
```

Dimana :

- Atribut `cx` dan `cy` mendefinisikan koordinat pusat lingkaran.
- Atribut `r` mendefinisikan jari-jari lingkaran.

Hasilnya :



3. Bentuk Elips

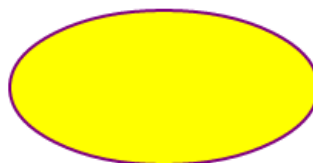
Elemen `<ellipse>` mendefinisikan pembuatan bentuk elips. Contoh kodenya :

```
<svg>
<ellipse cx="300" cy="80" rx="100" ry="50"
style="fill:yellow;stroke:purple;stroke-width:2"/>
</svg>
```

Dimana :

- Atribut `cx` mendefinisikan koordinat x pusat elips dan `cy` koordinat y pusat elips.
- Atribut `rx` mendefinisikan jari-jari horizontal dan `ry` mendefinisikan jari-jari vertical.

Hasilnya :



4. Bentuk Garis

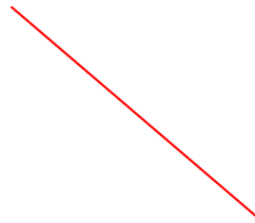
Elemen `<line>` mendefinisikan pembuatan garis. Contoh kode :

```
<svg>
  <line x1="0" y1="0" x2="200" y2="200"
        stroke:"rgb(255,0,0)" stroke-width:"2"/>
</svg>
```

Dimana :

- Atribut `x1` mendefinisikan koordinat x awal dari garis dan `y1` mendefinisikan koordinat y awal dari garis.
- Atribut `x2` mendefinisikan koordinat x akhir dari garis dan `y2` mendefinisikan koordinat y akhir dari garis.

Hasilnya :



5. Bentuk Poligon

Elemen `<polygon>` mendefinisikan sebuah polygon. Contoh kode :

```
<svg>
  <polygon points="200,10 250,190 160,210"
            fill:"lime" stroke:"purple" stroke-width:"1"/>
</svg>
```

Dimana :

- Atribut `points` mendefinisikan koordinat titik-titik sudut yang membentuk polygon itu.

Hasilnya :



6. Bentuk Polyline

Elemen `<polyline>` mendefinisikan sebuah polyline atau gabungan dari beberapa garis. Contoh kode :

```
<svg>
  <polyline points="0,40 40,40 40,80 80,80 80,120 120,120 120,160"
    "fill:"white" stroke:"red" stroke-width:"4"/>
</svg>
```

Dimana :

- Atribut `points` mendefinisikan koordinat masing-masing titik polyline tersebut.

Hasilnya :



7. Bentuk Path

Dalam path terdapat beberapa command untuk mendefinisikan path tersebut yaitu :

- M = moveto
- L = lineto
- H = horizontal lineto
- V = vertical lineto
- C = curveto
- S = smooth curveto
- Q = quadratic Bézier curve
- T = smooth quadratic Bézier curveto
- A = elliptical Arc
- Z = closepath

Dimana command-command tersebut dapat dituliskan dalam huruf besar (capital) atau kecil. Huruf besar berarti koordinat yang dipakai absolut dan huruf kecil koordinat yang dipakai relative. Contoh kode :

```
<svg>
  <path d="M150 0 L75 200 L225 200 Z" />
</svg>
```

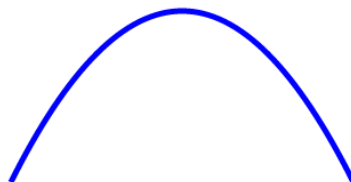
Kode diatas berarti mendefinisikan sebuah path dimana koordinat awal (M) adalah 150,0 kemudian dari titik awal tersebut dibuat garis (L) ke koordinat 75,200 kemudian dari koordinat itu dibuat garis (L) ke 225,200 dan path tersebut ditutup ke koordinat awal (Z). hasilnya :



Contoh kode :

```
<svg>
  <path d="M 100 350 q 150 -300 300 0" stroke="blue"
        stroke-width="5" fill="none" />
</svg>
```

Dimana kode diatas mendefinisikan path untuk membuat kurva, dimulai dari koordinat 100,350 kemudian 150,-300 merupakan control point (sebagai titik puncak kurva) dan 300,0 merupakan koordinat akhir kurva. Hasilnya adalah :



8. Bentuk Teks

Elemen <text> mendefinisikan pembuatan teks dengan svg. Contoh kode :

```
<svg>
  <text x="0" y="15" fill="red">I love SVG</text>
</svg>
```

Dimana :

- Atribut x dan y mendefinisikan koordinat dari teks tersebut.
- Fill mendefinisikan warna teks tersebut

Hasilnya :

I love SVG

SVG Gradient

Dalam SVG kita dapat memberi warna pada bentuk yang kita buat dengan gradasi atau perubahan warna dari satu warna ke warna lain. Ada dua tipe gradient dalam SVG yaitu Linear dan Radial.

1. Gradient Linear

Untuk membuat linear gradient didefinisikan dengan elemen `<linearGradient>` dimana elemen tersebut harus diberikan tag `<defs>`. Tag `<defs>` sendiri merupakan kependekan dari definition dimana mengandung definisi elemen khusus seperti gradient ini.

Linear Gradient dapat didefinisikan sebagai gradient horizontal, vertical dan menyudut. Dimana :

- Gradient horizontal dibuat saat y1 dan y2 sama koordinatnya dan x1 dan x2 berbeda koordinatnya.
- Gradient vertical dibuat saat x1 dan x2 sama koordinatnya dan y1 dan y2 berbeda koordinatnya.
- Gradient menyudut dibuat saat x1 dan x2 berbeda koordinatnya dan y1 dan y2 berbeda koordinatnya.

Contoh kode :

```
<svg>
  <defs>
    <linearGradient id="grad1" x1="0%" y1="0%" x2="100%" y2="0%">
      <stop offset="0%" style="stop-color:rgb(255,255,0);stop-opacity:1" />
      <stop offset="100%" style="stop-color:rgb(255,0,0);stop-opacity:1" />
    </linearGradient>
  </defs>
  <ellipse cx="200" cy="70" rx="85" ry="55" fill="url(#grad1)" />
</svg>
```

Pada kode diatas berarti kita mendefinisikan sebuah ellipse dengan gradient horizontal tipe vertical dari kuning ke merah.

Hasilnya :



2. Gradient Radial

Elemen `<radialGradient>` mendefinisikan gradient radial. Contoh kode :

```

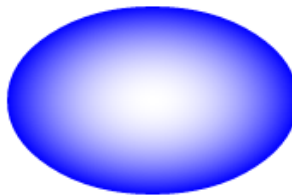
<svg>
  <defs>
    <radialGradient id="grad1" cx="50%" cy="50%" r="50%" fx="50%"
    fy="50%">
      <stop offset="0%" style="stop-color:rgb(255,255,255);
      stop-opacity:0" />
      <stop offset="100%" style="stop-color:rgb(0,0,255);stop-opacity:1"
    />
    </radialGradient>
  </defs>
  <ellipse cx="200" cy="70" rx="85" ry="55" fill="url(#grad1)" />
</svg>

```

Dimana :

- Atribut id dari tag <radialGradient> mendefinisikan nama untuk gradient tersebut.
- Atribut cx, cy, dan r mendefinisikan lingkaran terluar dan fx dan fy mendefinisikan lingkaran dalam.
- Jangkauan warna gradient dapat disusun dari dua atau lebih warna. Setiap warna diberikan tag <stop>. Atribut offset digunakan untuk mendefinisikan dimana gradient dimulai dan diakhiri.
- Atribut fill pada elemen ellipse menghubungkan dengan gradient.

Hasilnya :



SVG Transformasi

1. Membuat Grup

```

<svg>
  <g id="kotak">
    <rect x="100" y="100" width="100" height="50" fill="red"/>
    <rect x="200" y="100" width="100" height="50" fill="blue"/>
  </g>
  <use xlink:href="#kotak" x="250" y="0"/>
</svg>

```

Dimana :

-
- Kita misalnya membuat dua bentuk kotak yang akan dibuat grup.

- Tag `<g id="kotak">...</g>` merupakan definisi nama dari grup tersebut dengan nama kotak.
- Tag `<use xlink:href="kotak" x=" " y=" " />` untuk “memanggil” grup kotak tagi pada koordinat yang kita inginkan.

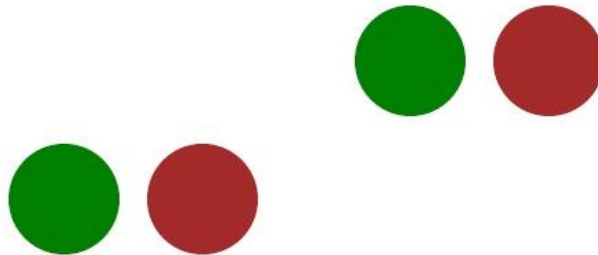
Hasilnya :



Atau dengan menggunakan command :

```
<svg>
  <defs>
    <g id="kotak">
      <rect x="100" y="100" width="100" height="50" fill="red" />
      <rect x="210" y="100" width="100" height="50" fill="blue" />
    </g>
    <g id="lingk">
      <circle cx="50" cy="80" r="40" fill="green" />
      <circle cx="150" cy="80" r="40" fill="brown" />
    </g>
  </defs>
  <use xlink:href="#lingk" x="250" y="0" />
  <use xlink:href="#lingk" x="0" y="100" />
</svg>
```

Hasilnya :

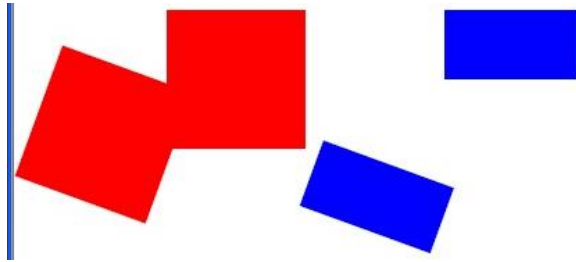


2. Rotasi

Untuk membuat rotasi digunakan `transform="rotate(x)"` dimana x merupakan besar sudut rotasi. Contoh kode :

```
<svg>
  <defs>
    <g id="kotak">
      <rect x="10" y="100" width="100" height="100" fill="red" />
      <rect x="210" y="100" width="100" height="50" fill="blue" />
    </g>
    <g id="lingk">
      <circle cx="50" cy="80" r="40" fill="green" />
      <circle cx="150" cy="80" r="40" fill="brown" />
    </g>
  </defs>
  <use xlink:href="#kotak" x="100" y="100" />
  <use xlink:href="#kotak" x="100" y="100" transform="rotate(20)" />
</svg>
```


Hasilnya :



3. Translasi

Untuk melakukan translasi digunakan `transform="translate(x,y)"` dimana `x,y` merupakan koordinat tujuan translasi. Contoh kode :

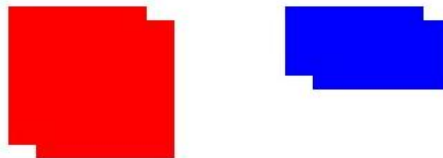
```
<svg>
<defs>
<g id="kotak">
<rect x="10" y="100" width="100" height="100" fill="red" />
<rect x="210" y="100" width="100" height="50" fill="blue" />
</g>

<g id="lingk">
<circle cx="50" cy="80" r="40" fill="green" />
<circle cx="150" cy="80" r="40" fill="brown" />
</g>
</defs>

<use xlink:href="#kotak" x="100" y="100" />
<use xlink:href="#kotak" x="100" y="100" transform="translate(20,10)" />

</svg>
```

Hasilnya :



4. Skala

Untuk melakukan penskalaan obyek digunakan `transform="scale(x)"` dimana `x` merupakan factor skala. Contoh kode :

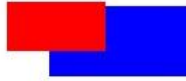
```
<svg>
<defs>
<g id="kotak">
<rect x="10" y="100" width="100" height="100" fill="red" />
<rect x="210" y="100" width="100" height="50" fill="blue" />
</g>

<g id="lingk">
<circle cx="50" cy="80" r="40" fill="green" />
<circle cx="150" cy="80" r="40" fill="brown" />
</g>
</defs>

<rect x="100" y="10" width="100" height="50" fill="blue" />
<rect x="100" y="10" width="100" height="50" fill="red" transform="scale(0.7)" />

</svg>
```

Hasilnya :



Dimana pada kedua contoh diatas gambar kotak biru merupakan kotak awal sebelum dilakukan skala dan kotak merah merupakan hasil skala.

5. Skew

Skew ini dapat dilakukan terhadap sumbu x maupun y. Untuk melakukan skew digunakan `transform="skew(x)"`. contoh kodenya :

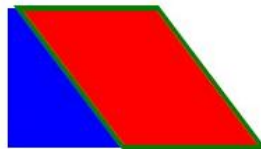
- Skew terhadap sumbu-x

```
<svg>
<defs>
<g id="kotak">
<rect x="10" y="100" width="100" height="100" fill="red"/>
<rect x="210" y="100" width="100" height="50" fill="blue"/>
</g>

<g id="lingk">
<circle cx="50" cy="80" r="40" fill="green"/>
<circle cx="150" cy="80" r="40" fill="brown"/>
</g>
</defs>

<rect x="100" y="10" width="100" height="100" fill="blue"/>
<rect x="100" y="10" width="100" height="100" fill="red" stroke="green" stroke-width="4" transform="skewX(37)"/>
</svg>
```

Hasilnya :



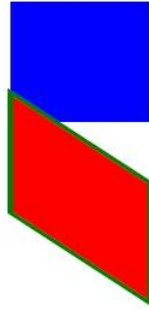
- Skew terhadap sumbu-y

```
<svg>
<defs>
<g id="kotak">
<rect x="10" y="100" width="100" height="100" fill="red"/>
<rect x="210" y="100" width="100" height="50" fill="blue"/>
</g>

<g id="lingk">
<circle cx="50" cy="80" r="40" fill="green"/>
<circle cx="150" cy="80" r="40" fill="brown"/>
</g>
</defs>

<rect x="100" y="10" width="100" height="100" fill="blue"/>
<rect x="100" y="10" width="100" height="100" fill="red" stroke="green" stroke-width="4" transform="skewY(37)"/>
</svg>
```

Hasilnya :



Dimana pada kedua contoh diatas gambar kotak biru merupakan kotak awal sebelum dilakukan skew dan obyek merah merupakan hasil skew.