

SVG-Kurzreferenz

1. Allgemeines

- `<?xml ... ?>`

XML-Verarbeitungsanweisung

`version` ist immer 1.0

`encoding` Zeichencodierung (üblicherweise `utf-8` oder `latin-1`)

- `<svg> </svg>`

Wurzelement einer SVG-Grafik

`width` Breite der Grafik

`height` Höhe der Grafik

`xmlns` Namensraum von SVG ("`http://www.w3.org/2000/svg`")

- `rgb(rot, grün, blau)`

Definiert eine Farbe durch ihren Rot-Grün-Blau-Anteil. Diese Anteile können entweder als ganze Zahl (0–255) oder als Prozentwert (0%–100%) angegeben werden.

Ab SVG-Version 1.1 können folgende vordefinierte Farbnamen verwendet werden:

AliceBlue, AntiqueWhite, Aqua, Aquamarine, Azure, Beige, Bisque, Black, BlanchedAlmond, Blue, BlueViolet, Brown, BurlyWood, CadetBlue, Chartreuse, Chocolate, Coral, Cornflower, Cornsilk, Crimson, Cyan, DarkBlue, DarkCyan, DarkGoldenrod, DarkGray, DarkGreen, DarkKhaki, DarkMagenta, DarkOliveGreen, DarkOrange, DarkOrchid, DarkRed, DarkSalmon, DarkSeaGreen, DarkSlateBlue, DarkSlateGray, DarkTurquoise, DarkViolet, DeepPink, DeepSkyBlue, DimGray, DodgerBlue, Firebrick, FloralWhite, ForestGreen, Fuchsia, Gainsboro, GhostWhite, Gold, Goldenrod, Gray, Grey, Green, GreenYellow, Honeydew, HotPink, IndianRed, Indigo, Ivory, Khaki, Lavender, LavenderBlush, LawnGreen, LemonChiffon, LightBlue, LightCoral, LightCyan, LightGoldenrod, LightGreen, LightGrey, LightPink, LightSalmon, LightSeaGreen, LightSkyBlue, LightSlateGray, LightSteelBlue, LightYellow, Lime, LimeGreen, Linen, Magenta, Maroon, MediumAquamarine, MediumBlue, MediumOrchid, MediumPurple, MediumSeaGreen, MediumSlateBlue, MediumSpringGreen, MediumTurquoise, MediumVioletRed, MidnightBlue, MintCream, MistyRose, Moccasin, NavajoWhite, Navy, OldLace, Olive, OliveDrab, Orange, OrangeRed, Orchid, PaleGoldenrod, PaleGreen, PaleTurquoise, PaleVioletRed, PapayaWhip, PeachPuff, Peru, Pink, Plum, PowderBlue, Purple, Red, RosyBrown, RoyalBlue, SaddleBrown, Salmon, SandyBrown, SeaGreen, Seashell, Sienna, Silver, SkyBlue, SlateBlue, SlateGray, Snow, SpringGreen, SteelBlue, Tan, Teal, Thistle, Tomato, Turquoise, Violet, Wheat, White, WhiteSmoke, Yellow, YellowGreen,

- Masseneinheiten werden unmittelbar hinter eine Masszahl geschrieben.

`px` Pixel: eine Einheit in lokalen Koordinaten

`em` Geviertbreite und -höhe: Schriftgröße der aktuellen Schriftart

`ex` Die Höhe des Buchstabens x in der aktuellen Schriftart

`pt` Punkt (*point*): 0.3527mm oder 1/72 Zoll

`pc` Pica: 4.23mm oder 12pt

`mm` Millimeter

`cm` Zentimeter

`in` Zoll (*inch*): 25.4mm

- x - und y -Koordinaten können entweder durch ein Komma oder ein Leerzeichen getrennt angegeben werden. Die einzelnen Koordinatenpaare werden durch ein oder mehrere Leerzeichen getrennt. Als Leerzeichen im weiteren Sinne gelten auch Zeilenschaltungen und Tabulatoren.

Eine positive x -Koordinate bezeichnet den Abstand vom linken Bildschirmrand. Eine positive y -Koordinate bezeichnet den Abstand vom oberen Bildschirmrand.

- `<title></title>`

Enthält den Titel eines SVG-Elements. Praktisch jedes SVG-Element kann einen Titel als Kindelement enthalten.

- `<desc></desc>`

Beschreibung (description) eines SVG-Elements. Praktisch jedes SVG-Element kann eine Beschreibung als Kindelement enthalten.

- `<g> </g>`

Mit diesem Element (group) können mehrere grafische Elemente zu einer Einheit zusammengefasst werden. Auf diese Weise lassen sich verschiedene Eigenschaften (Linienfarbe, Titel, Transformationen, ...) auf die Kindelemente übertragen.

2. Grundformen

`fill` Füllfarbe

`stroke` Linienfarbe

- `<circle />`

Zeichnet einen Kreis.

`cx` definiert die x -Koordinate des Kreismittelpunkts

`cy` definiert die y -Koordinate des Kreismittelpunkts

`r` definiert den Radius des Kreises

- `<ellipse />`

Zeichnet eine Ellipse.

`cx` definiert die x -Koordinate des Ellipsenmittelpunkts

`cy` definiert die y -Koordinate des Ellipsenmittelpunkts

`rx` definiert die Länge der horizontalen Halbachse

`ry` definiert die Länge der vertikalen Halbachse

- `<line />`

Zeichnet eine Strecke.

`x1` definiert die x -Koordinate des einen Endpunkts

`y1` definiert die y -Koordinate des einen Endpunkts

x2 definiert die x -Koordinate des anderen Endpunkts

y2 definiert die y -Koordinate des anderen Endpunkts

- `<polygon />`

Zeichnet ein Polygon.

points Eine Liste mit xy -Koordinaten

- `<polyline />`

Zeichnet einen (offenen) Polygonzug.

points Eine Liste mit xy -Koordinaten

- `<rect />`

Zeichnet ein Rechteck.

x definiert die Position des Rechtecks vom linken Rand.

y definiert die Position des Rechtecks vom oberen Rand

width definiert die Breite des Rechtecks

height definiert die Höhe des Rechtecks

3. Text

`<text> </text>`

Gibt einen Text aus.

x x -Koordinate des Textes

y y -Koordinate des Textes

font-size Grösse des Zeichensatzes

font-family Zeichensatzfamilie (`serif`, `sans-serif`, `mono`)

font-weight Stärke der Zeichen (`normal`, `bold`)

rotate Drehung der Buchstaben um den Wert des angegebenen Winkels (im Uhrzeigersinn)

4. Pfade

`<path />`

Erzeugt einen Pfad, der gezeichnet oder gefüllt werden kann.

d Eine Zeichenkette, die aus Kommandos und Koordinaten besteht (siehe unten).

Kommandos werden in grossen oder kleinen Buchstaben angegeben. Grosse Buchstaben entsprechen absoluten Koordinatenangaben. Kleine Buchstaben entsprechen relativen Angaben, relativ zum letzten Punkt des Pfades (also wirklich Punkt und nicht Kontrollpunkt).

- M, m (x y)+ Legt die Position des Zeichenstiftes fest, x für waagrecht und y für senkrecht. Es folgt mindestens ein Koordinatenpaar, bei mehreren wird ein Linienzug mit geraden Verbindungslinien angenommen.
- L, l (x y)+ Zeichnet eine gerade Linie von der aktuellen Position zu den angegebenen (x,y) Koordinaten, welche die neue Position festlegen. Weitere Koordinatenpaare können angegeben werden, das ergibt dann einen Linienzug.
- H, h x + Zeichnet eine horizontale Linie von der aktuellen Position, die vertikale Position bleibt gleich. Weitere Koordinaten führen wieder zu einem Linienzug.
- V, v y + Zeichnet eine vertikale Linie von der aktuellen Position, die horizontale Position bleibt gleich. Weitere Koordinaten führen wieder zu einem Linienzug.
- C, c ($x1$ $y1$ $x2$ $y2$ x y)+ Zeichnet eine kubische Bézierkurve von der aktuellen Position nach (x,y) unter der Verwendung von ($x1,y1$) als dem Kontrollpunkt am Anfang der Kurve und ($x2,y2$) als dem Kontrollpunkt am Ende der Kurve. Angabe von weiteren Kontrollpunkten und Punkten führt zu einem Kurvenzug. Die Anzahl der (Kontroll-)Punkte muss immer exakt stimmen.
- S, s ($x2$ $y2$ x y)+ Aus dem letzten Kontrollpunkt und Punkt des vorherigen kubischen Bézierkurvensegmentes wird eine stetig differenzierbare Fortsetzung des Pfades berechnet. Ist das vorherige Segment keine kubische Bézierkurve, so wird der fehlende Kontrollpunkt als identisch mit dem Ausgangspunkt angenommen. Angabe von weiteren Kontrollpunkten und Punkten führt zu einem Kurvenzug. Die Anzahl der (Kontroll-)Punkte muss immer exakt stimmen.
- Q, q ($x1$ $y1$ x y)+ Zeichnet eine quadratische Bézierkurve von der aktuellen Position nach (x,y) unter Verwendung von ($x1,y1$) als Kontrollpunkt. Angabe von weiteren Kontrollpunkten und Punkten führt zu einem Kurvenzug. Die Anzahl der (Kontroll-)Punkte muss immer exakt stimmen.
- T, t (x y)+ Aus dem letzten Kontrollpunkt und Punkt des vorherigen quadratischen Bézierkurvensegmentes wird eine stetig differenzierbare Fortsetzung des Pfades berechnet. Ist das vorherige Segment keine quadratische Bézierkurve, so wird der fehlende Kontrollpunkt als identisch mit dem Ausgangspunkt angenommen. Angabe von weiteren Kontrollpunkten und Punkten führt zu einem Kurvenzug.
- A, a (rx ry *x-axis-rotation* *large-arc-flag* *sweep-flag* x y)+ Zeichnet einen elliptischen Kreisbogen von der aktuellen Position nach (x, y). Die Grösse und die Ausrichtung der Ellipse werden bestimmt durch zwei Radien (rx, ry) und einer x -Achsen-Rotation, die bestimmt, wie die gesamte Ellipse relativ zum Koordinatensystem gedreht wird. Da es dann noch die Möglichkeit gibt, wie herum der Bogen gemalt wird und ob ein kleiner oder grosser Bogen gemalt wird, gibt es auch noch Boolesche Parameter *large-arc-flag* *sweep-flag*. Die Werte sind 0 oder 1. *large-arc-flag* 1 bewirkt, dass der grosse Bogen gemalt wird, der Wert 0 entspricht dem kleinen Bogen. *sweep-flag* 1 entspricht einer positiven Malrichtung,

0 einer negativen. Im Bedarfsfalle, wenn es keinen Ellipsenbogen gibt, der nach den Angaben Anfangs- und Endpunkt verbindet, werden die Angaben vom Darstellungsprogramm umskaliert.

Z, z (**keine**) Zeichnet eine Linie von der aktuellen Position zur ersten Koordinate des aktuellen (Unter-)Pfades.

5. Transformationen

Mit dem Attribut `transform` können grafische Elemente und gruppierte Elemente geometrisch transformiert werden.

Es können mehrere Transformationen hintereinander ausgeführt werden. Man beachte, dass dieselben Transformationen in einer anderen Reihenfolge in der Regel eine andere Wirkung haben.

- `translate(dx, dy)` Verschiebung um dx in x -Richtung und um dy in y -Richtung.
- `scale(fx, fy)` Skalierung (Streckung/Stauchung) um den Faktor fx in x -Richtung und um den Faktor fy in y -Richtung
- `rotate(winkel, x, y)` Drehung mit Winkel $\langle\text{winkel}\rangle$ im Uhrzeigersinn um das Zentrum mit den Koordinaten x und y . Ohne Angabe eines Drehzentrums, wird der Ursprung angenommen.
- `skewX(winkel)` Scherung in x -Richtung um den Winkel $winkel$.
- `skewY(winkel)` Scherung in y -Richtung um den Winkel $winkel$.

6. Animationen

<animate>

Animiert SVG-Objekte

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN"
"http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">
```

```
<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink">
```

```
<circle cx="50" cy="50" r="20">
  <animate
    attributeName="cx"
    from="0"
    to="98"
    dur="10s"
    repeatCount="indefinite"/>
</circle>
```

```
</svg>
```

7. Elementreferenz

<http://www.w3.org/TR/SVG11/eltindex.html>