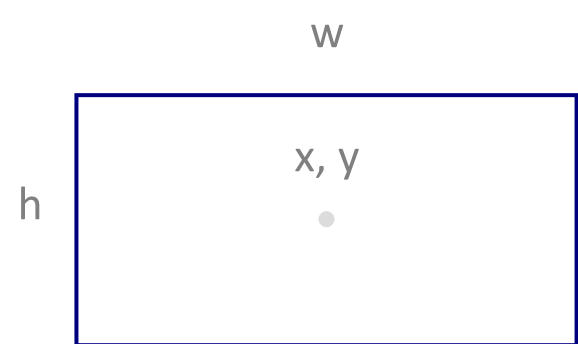


# SVG Go Library

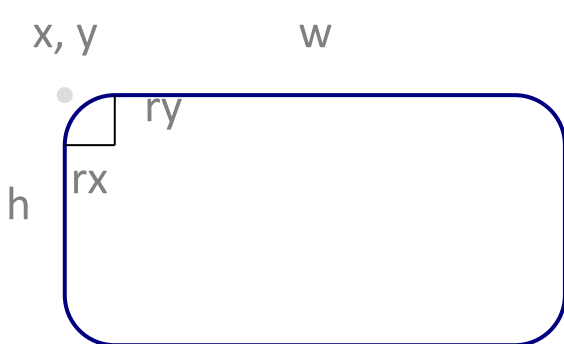
github.com/ajstarks/svgo



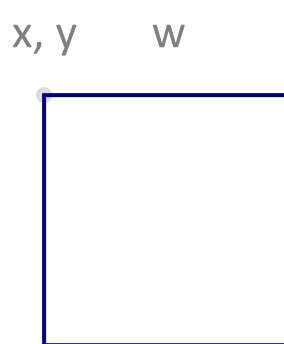
Rect(x, y, w, h int, style ...string)



CenterRect(x, y, w, h int, style ...string)



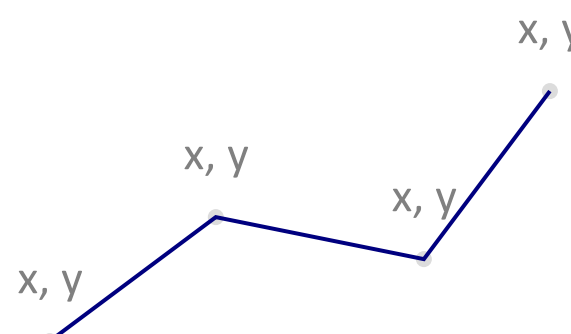
Roundrect(x, y, w, h, rx, ry int, style ...string)



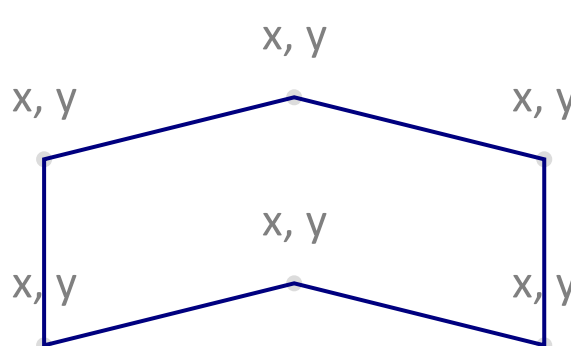
Square(x, y, w int, style ...string)



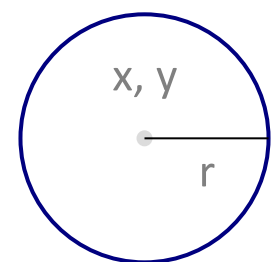
Line(x1, y1, x2, y2 int, style ...string)



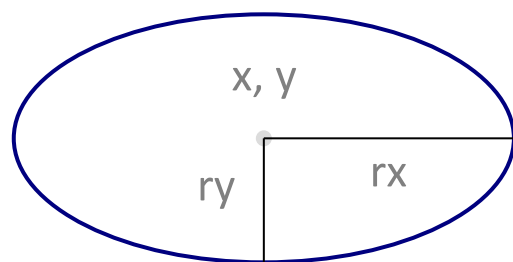
Polyline(x, y []int, style ...string)



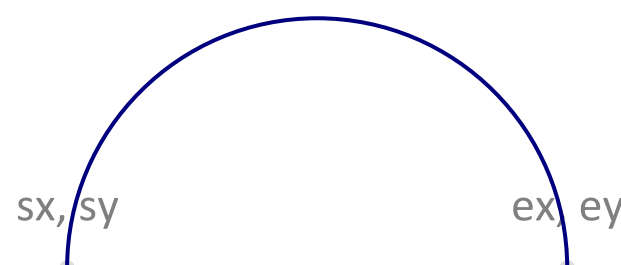
Polygon(x, y []int, style ...string)



Circle(x, y, r int, style ...string)



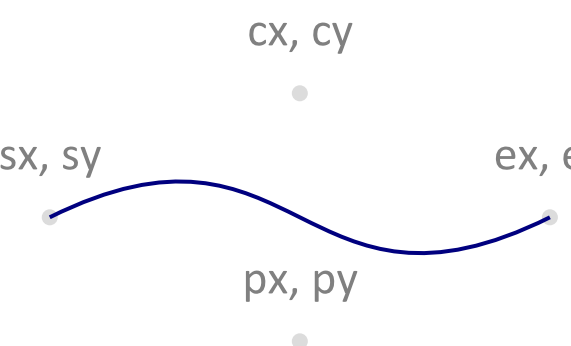
Ellipse(x, y, rx, ry int, style ...string)



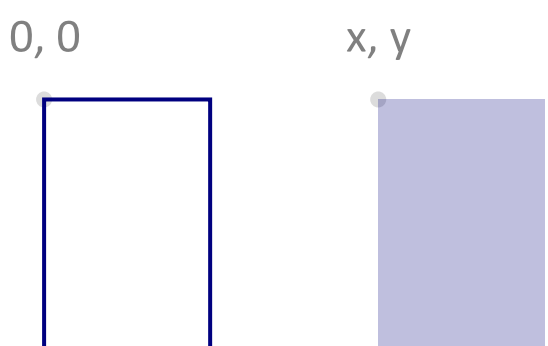
Arc(sx, sy, ax, ay, r int, lflag, sflag bool, ex, ey int, style ...string)



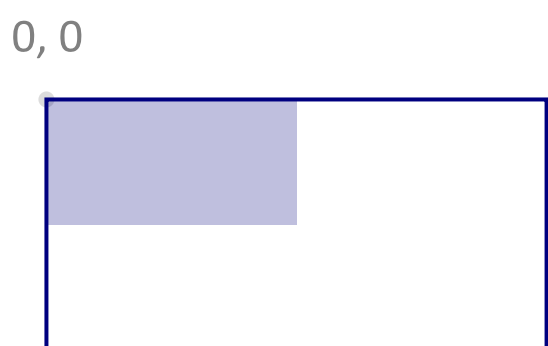
Qbez(sx, sy, cx, cy, ex, ey int, style ...string)



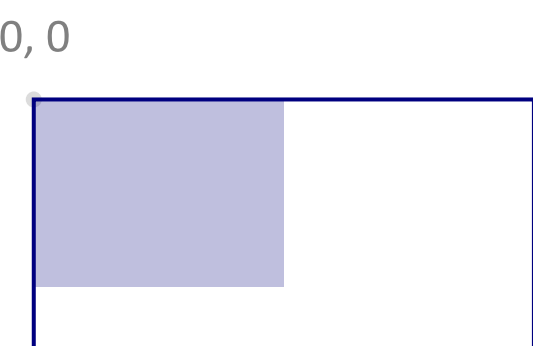
Bezier(sx, sy, cx, cy, px, py, ex, ey int, style ...string)



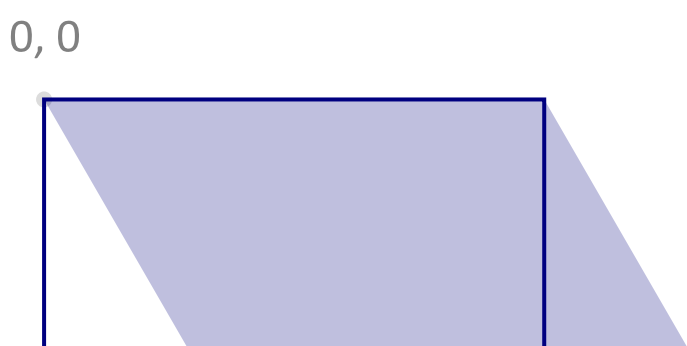
Translate(x, y int)



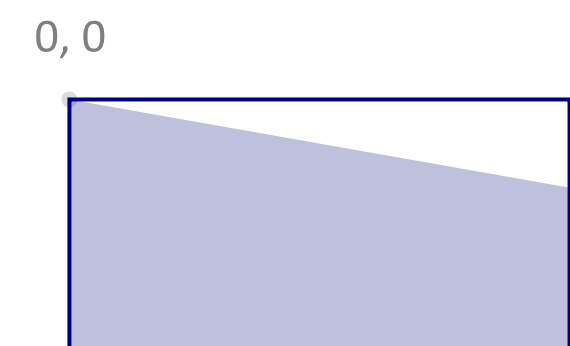
Scale(n float64)



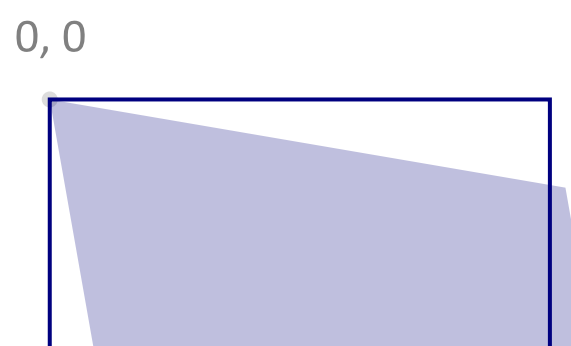
ScaleXY(x, y float64)



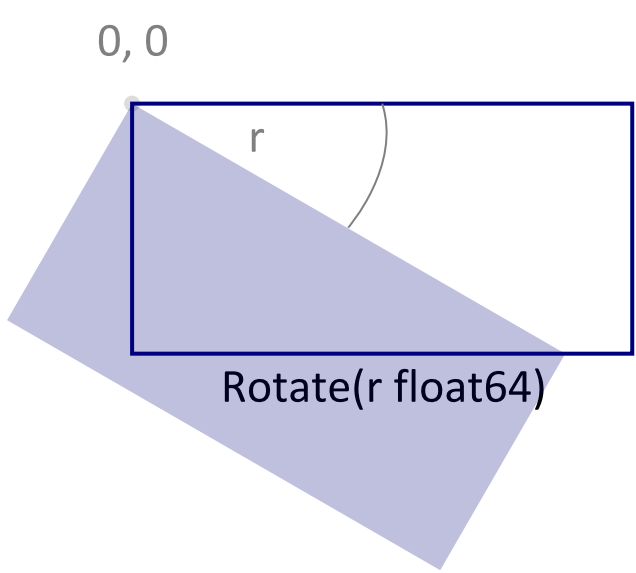
SkewX(a float64)



SkewY(a float64)



SkewXY(x, y float64)



Rotate(r float64)

hello, this is SVG

(x, y)

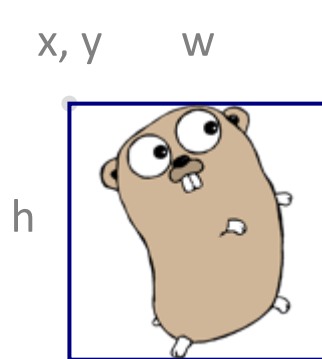
Text(x, y int, s string, style ...string)

It's "fine" & "dandy" to draw text along a path

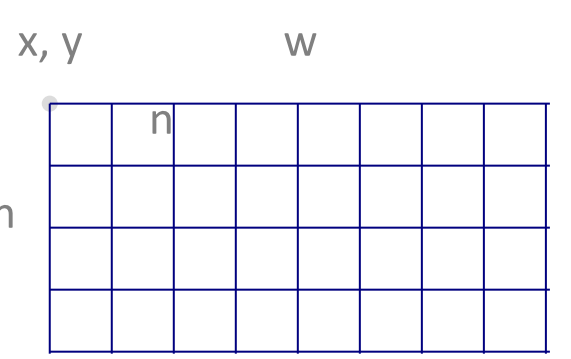
Textpath(s, pathid string, style ...string)



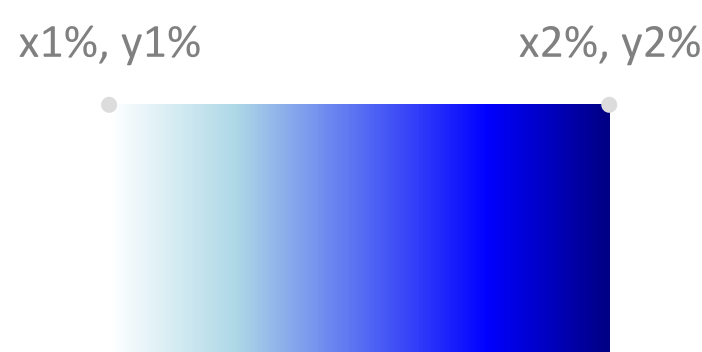
Path(s string, style ...string)



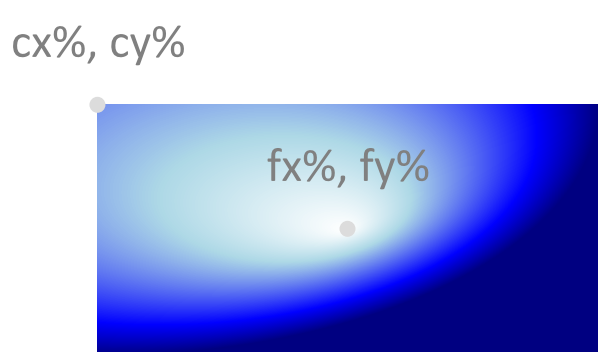
Image(x, y, w, h, int path string, style ...string)



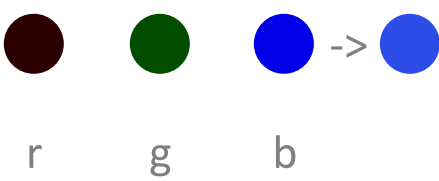
Grid(x, y, w, h, n int, style ...string)



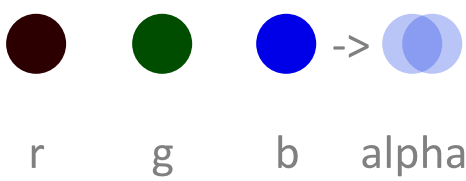
LinearGradient(s string, x1, y1, x2, y2 uint8, oc []Offcolor)



RadialGradient(s string, cx, cy, r, fx, fy uint8, oc []Offcolor)



RGB(r, g, b int)



RGBA(r, g, b int, opacity float64)

New(w io.Writer)  
Start(w, h int, options ...string)/End()  
Startview(w, h, minx, miny, vw, vh int)  
Group(s ...string)/Gend()  
Gstyle(s string)/Gend()  
Gtransform(s string)/Gend()  
Gid(id string)/Gend()  
ClipPath(s ...string)/ClipEnd()  
Def()/DefEnd()  
Marker()/MarkerEnd()  
Desc(s string)  
Title(s string)  
Script(type, data ...string)  
Mask(id string, x, y, w, h int, style ...string)/MaskEnd()  
Link(href string, title string)/LinkEnd()  
Use(x int, y int, link string, style ...string)

specify destination  
begin/end the document  
begin/end the document with viewport  
begin/end group with attributes  
begin/end group style  
begin/end group transform  
begin/end group id  
begin/end clip path  
begin/end a definition block  
begin/end markers  
set the description element  
set the title element  
define a script  
begin/end mask element  
begin/end link to href, with a title  
use defined objects

Textlines(x, y int, s []string, size, spacing int, fill, align string)