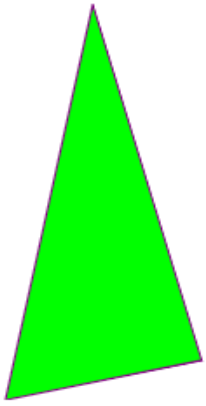
06. SVG Polygon & Path.

The **<polygon>** element is used to create a graphic that contains at least three sides.

Polygons are made of straight lines, and the shape is "closed" (all the lines connect up).

6.1. Example 1

The following example creates a polygon with three sides:



Example: Here is the SVG code:

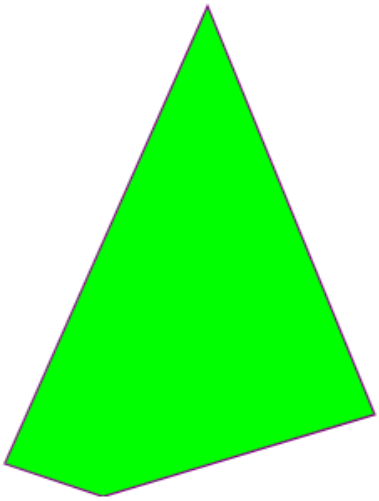
```
<svg height="210" width="500">
  <polygon points="200,10 250,190 160,210"
style="fill:lime;stroke:purple;stroke-width:1" />
</svg>
```

Code explanation:

- The points attribute defines the x and y coordinates for each corner of the polygon

6.2. Example 2

The following example creates a polygon with four sides:

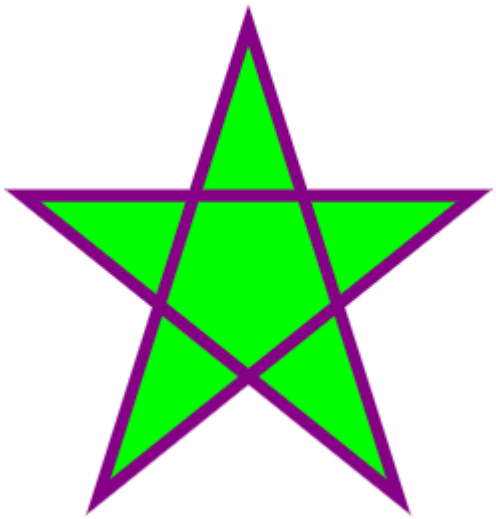


Example: Here is the SVG code:

```
<svg height="250" width="500">
  <polygon points="220,10 300,210 170,250 123,234"
style="fill:lime;stroke:purple;stroke-width:1" />
</svg>
```

6.3. Example 3

Use the <polygon> element to create a star:

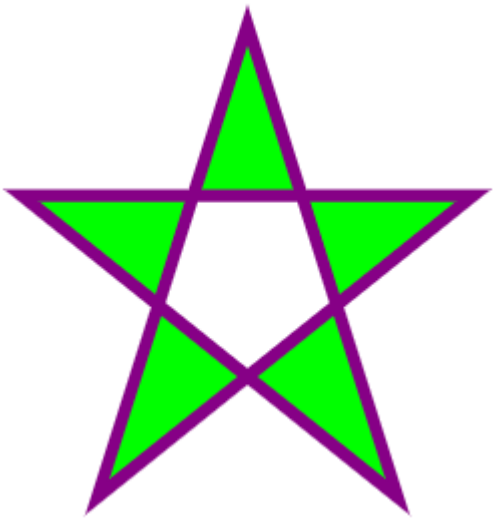


Example: Here is the SVG code:

```
<svg height="210" width="500">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
  style="fill:lime;stroke:purple;stroke-width:5;fill-rule:nonzero;" /
>
</svg>
```

6.4. Example 4

Change the fill-rule property to "evenodd":



Example: Here is the SVG code:

```
<svg height="210" width="500">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
  style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;" /
>
</svg>
```

6.5. SVG Path

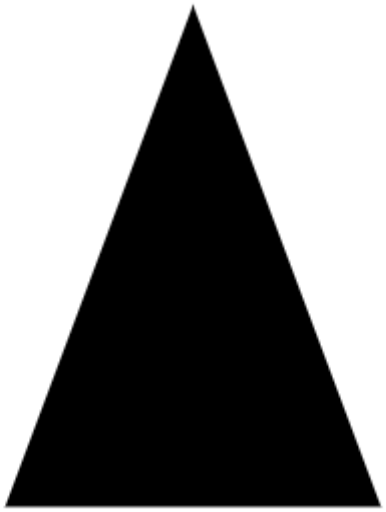
The **<path>** element is used to define a path. The following commands are available for path data:

- 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

Note: All of the commands above can also be expressed with lower letters. Capital letters means absolutely positioned, lower cases means relatively positioned.

6.6. Example 5

The example below defines a path that starts at position 150,0 with a line to position 75,200 then from there, a line to 225,200 and finally closing the path back to 150,0:



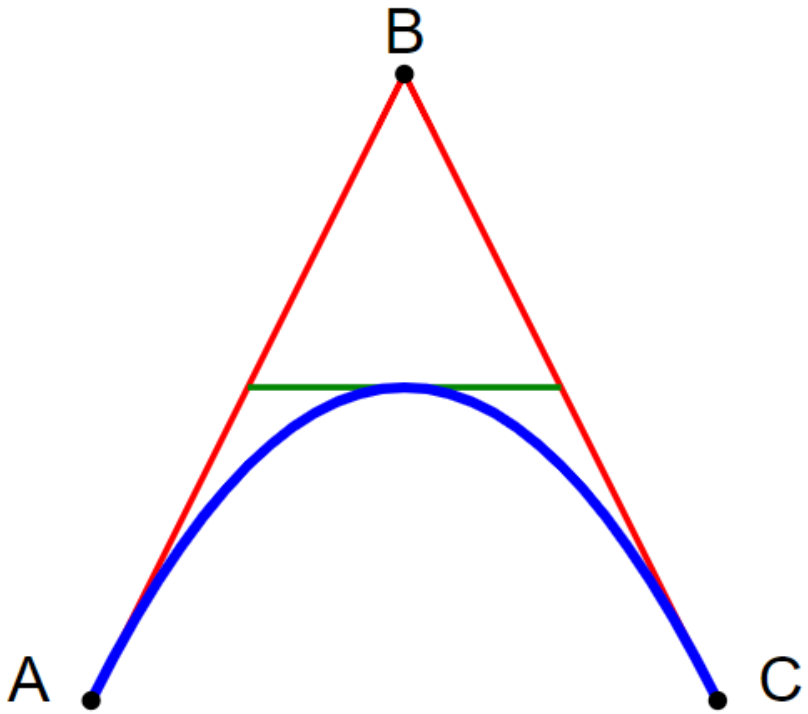
Example: Here is the SVG code:

```
<svg height="210" width="400">
  <path d="M150 0 L75 200 L225 200 Z" />
</svg>
```

6.7. Example 6

Bézier curves are used to model smooth curves that can be scaled indefinitely. Generally, the user selects two endpoints and one or two control points. A Bézier curve with one control point is called a quadratic Bézier curve and the kind with two control points is called cubic.

The following example creates a quadratic Bézier curve, where A and C are the start and end points, B is the control point:



Example: Here is the SVG code:

```
<svg height="400" width="450">
  <path id="lineAB" d="M 100 350 l 150 -300" stroke="red"
  stroke-width="3" fill="none" />
  <path id="lineBC" d="M 250 50 l 150 300" stroke="red"
  stroke-width="3" fill="none" />
  <path d="M 175 200 l 150 0" stroke="green" stroke-width="3"
  fill="none" />
  <path d="M 100 350 q 150 -300 300 0" stroke="blue"
```

```
    stroke-width="5" fill="none" />
    <!-- Mark relevant points -->
    <g stroke="black" stroke-width="3" fill="black">
      <circle id="pointA" cx="100" cy="350" r="3" />
      <circle id="pointB" cx="250" cy="50" r="3" />
      <circle id="pointC" cx="400" cy="350" r="3" />
    </g>
    <!-- Label the points -->
    <g font-size="30" font-family="sans-serif" fill="black"
  stroke="none"
    text-anchor="middle">
      <text x="100" y="350" dx="-30">A</text>
      <text x="250" y="50" dy="-10">B</text>
      <text x="400" y="350" dx="30">C</text>
    </g>
  </svg>
```

Complex? YES!!!! Because of the complexity involved in drawing paths it is highly recommended to use an SVG editor to create complex graphics.