

Review

```
var someVariable = 23423;
```

```
var someVariable = "Hello World";
```

```
var someVariable = false;
```

```
var someVariable = [55, 11, "11", true];
```

```
var someVariable = {name: "Sarah", age: 27};
```

```
var someVariable = function(arg){ return arg; }
```

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,   color: "#ff0000"},  
  {name: "Orange",  fruit: true,   color: "#ff7f00"},  
  {name: "Banana",   fruit: true,   color: "#ffff00"},  
  {name: "Cucumber", fruit: false,  color: "#00ff00"}  
];
```

What's the notation to find out whether a cucumber is a fruit?

```
food[3].fruit  
food[3]["fruit"]
```

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,  color: "#ff0000"},  
  {name: "Orange",  fruit: true,  color: "#ff7f00"},  
  {name: "Banana",   fruit: true,  color: "#ffff00"},  
  {name: "Cucumber", fruit: false, color: "#00ff00"}  
];
```

What's the notation to find out the color of a banana?

```
food[2].color  
food[2]["color"]
```

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,   color: "#ff0000"},  
  {name: "Orange",  fruit: true,   color: "#ff7f00"},  
  {name: "Banana",  fruit: true,   color: "#ffff00"},  
  {name: "Cucumber", fruit: false,  color: "#00ff00"}  
];
```

What's the notation to find out the name of the apple?

```
food[0].name  
food[0]["name"]
```

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,   color: "#ff0000"},  
  {name: "Orange",  fruit: true,   color: "#ff7f00"},  
  {name: "Banana",  fruit: true,   color: "#ffff00"},  
  {name: "Cucumber", fruit: false,  color: "#00ff00"}  
];
```

What's the notation to get back the entire object of the orange?

```
food[1]  
food[1]
```

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,  color: "#ff0000"},  
  {name: "Orange",  fruit: true,  color: "#ff7f00"},  
  {name: "Banana",  fruit: true,  color: "#ffff00"},  
  {name: "Cucumber", fruit: false, color: "#00ff00"}  
];  
  
var colors = food.map(function(d){ return d.color; });
```

What is the value of the colors variable?

```
["#ff0000", "#ff7f00", "#ffff00", "#00ff00"]
```

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,   color: "#ff0000"},  
  {name: "Orange",  fruit: true,   color: "#ff7f00"},  
  {name: "Banana",  fruit: true,   color: "#ffff00"},  
  {name: "Cucumber", fruit: false,  color: "#00ff00"}  
];  
  
var colors = food.map(function(d) { return d.color; });
```

Review Question

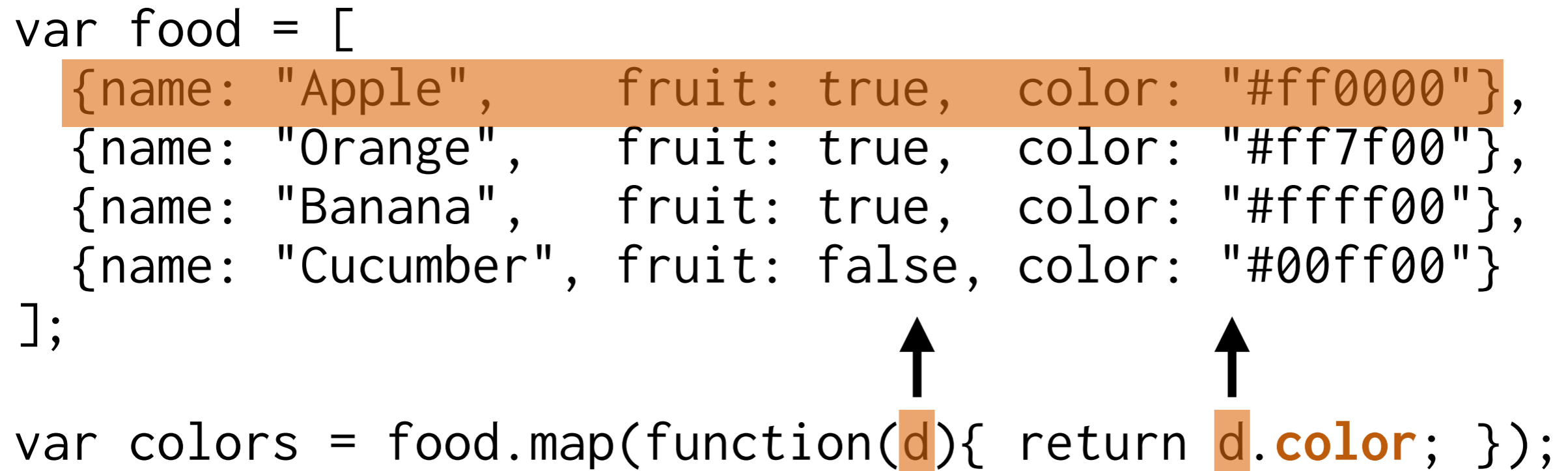
```
var food = [  
  {name: "Apple",    fruit: true,   color: "#ff0000"},  
  {name: "Orange",  fruit: true,   color: "#ff7f00"},  
  {name: "Banana",  fruit: true,   color: "#ffff00"},  
  {name: "Cucumber", fruit: false,  color: "#00ff00"}  
];  
  
var colors = food.map(function(d){ return d.color; });
```



run a function for each item in the array

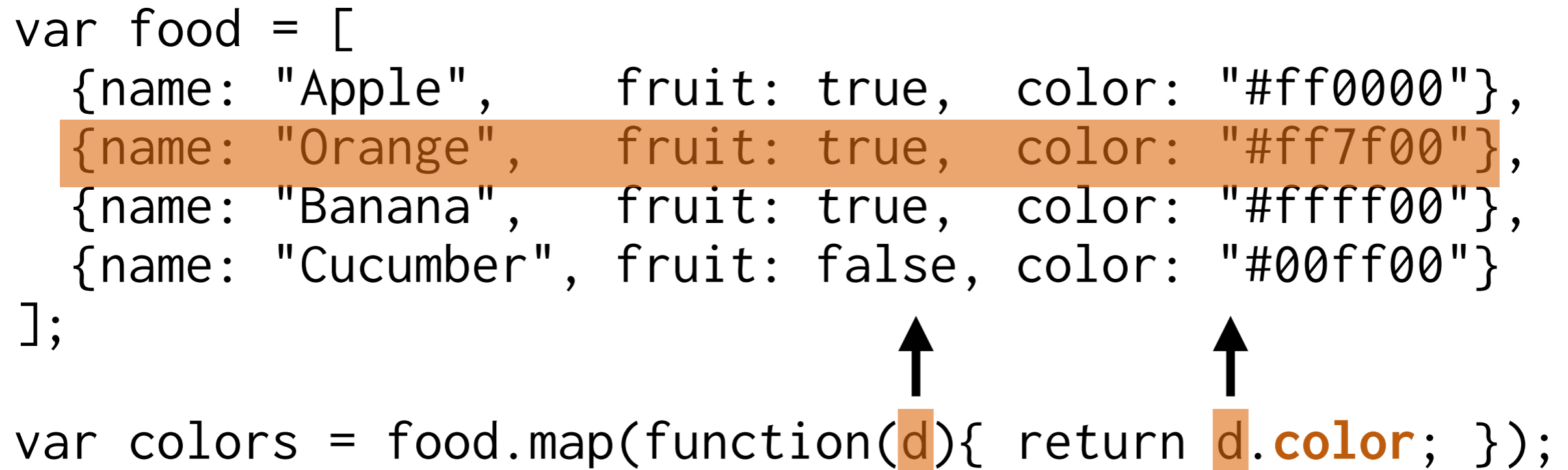
Review Question

```
var food = [  
  {name: "Apple", fruit: true, color: "#ff0000"},  
  {name: "Orange", fruit: true, color: "#ff7f00"},  
  {name: "Banana", fruit: true, color: "#ffff00"},  
  {name: "Cucumber", fruit: false, color: "#00ff00"}  
];  
  
var colors = food.map(function(d) { return d.color; });
```




Review Question

```
var food = [  
  {name: "Apple",    fruit: true,  color: "#ff0000"},  
  {name: "Orange",  fruit: true,  color: "#ff7f00"},  
  {name: "Banana",  fruit: true,  color: "#ffff00"},  
  {name: "Cucumber", fruit: false, color: "#00ff00"}  
];  
  
var colors = food.map(function(d) { return d.color; });
```



Review Question

```
var food = [  
  {name: "Apple",    fruit: true,  color: "#ff0000"},  
  {name: "Orange",  fruit: true,  color: "#ff7f00"},  
  {name: "Banana",  fruit: true,  color: "#ffff00"},  
  {name: "Cucumber", fruit: false, color: "#00ff00"}  
];  
  
var colors = food.map(function(d) { return d.color; });
```




Review Question

```
var food = [  
  {name: "Apple",    fruit: true,  color: "#ff0000"},  
  {name: "Orange",  fruit: true,  color: "#ff7f00"},  
  {name: "Banana",  fruit: true,  color: "#ffff00"},  
  {name: "Cucumber", fruit: false, color: "#00ff00"}  
];  
  
var colors = food.map(function(d) { return d.color; });
```

The diagram illustrates the variable resolution in the provided code. Two arrows point from the variable `d` in the function parameter and the property access `d.color` in the return statement to the corresponding elements in the `Cucumber` object of the `food` array above.

Review Question

```
var food = [  
  {name: "Apple",    fruit: true,   color: "#ff0000"},  
  {name: "Orange",  fruit: true,   color: "#ff7f00"},  
  {name: "Banana",  fruit: true,   color: "#ffff00"},  
  {name: "Cucumber", fruit: false,  color: "#00ff00"}  
];  
  
var colors = food.map(function(d) { return d.color; });
```



SVG

Scalable Vector Graphics

RASTER VS. VECTOR



Raster (Bitmap)



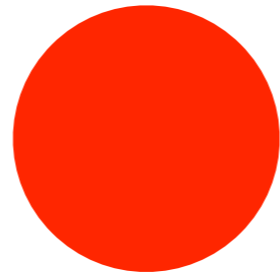
Vector (Graphic)

First understand SVG

`<svg>`

`</svg>`

First understand SVG

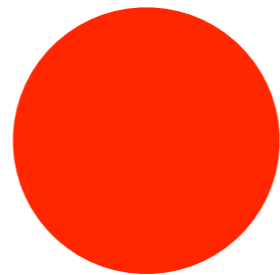


```
<svg>
```

```
<circle cx="0" cy="0" r="10" fill="red"></circle>
```

```
</svg>
```

First understand SVG



Hello

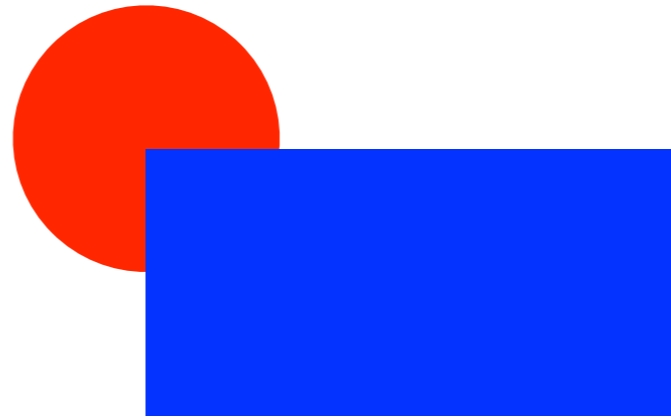
```
<svg>
```

```
<circle cx="0" cy="0" r="10" fill="red"></circle>
```

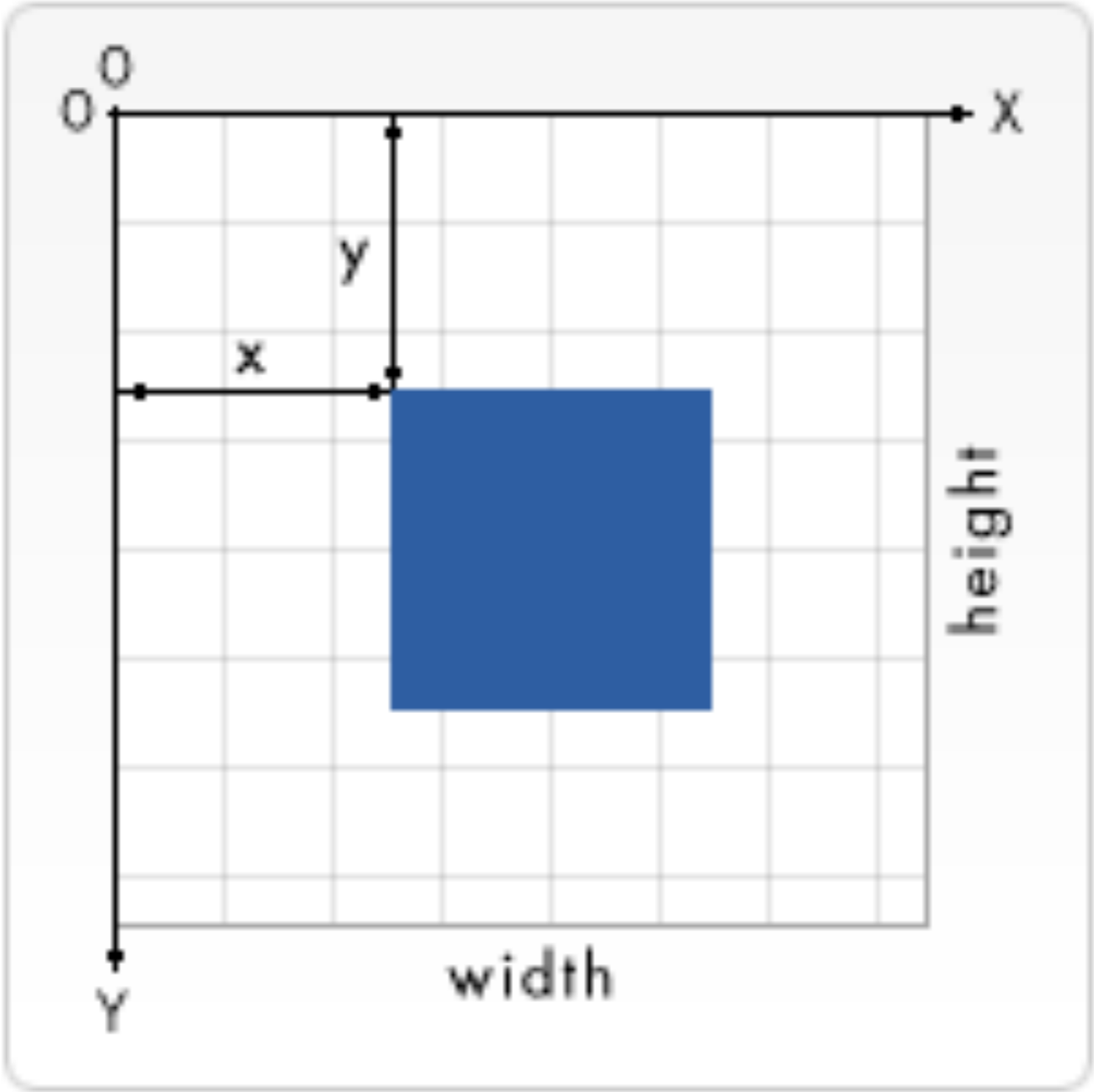
```
<text x="0" y="10">Hello</text>
```

```
</svg>
```

First understand SVG



```
<svg>  
  <circle cx="0" cy="0" r="10" fill="red"></circle>  
  <text x="0" y="10">Hello</text>  
  <rect width="30" height="10" style="fill:blue"></rect>  
</svg>
```



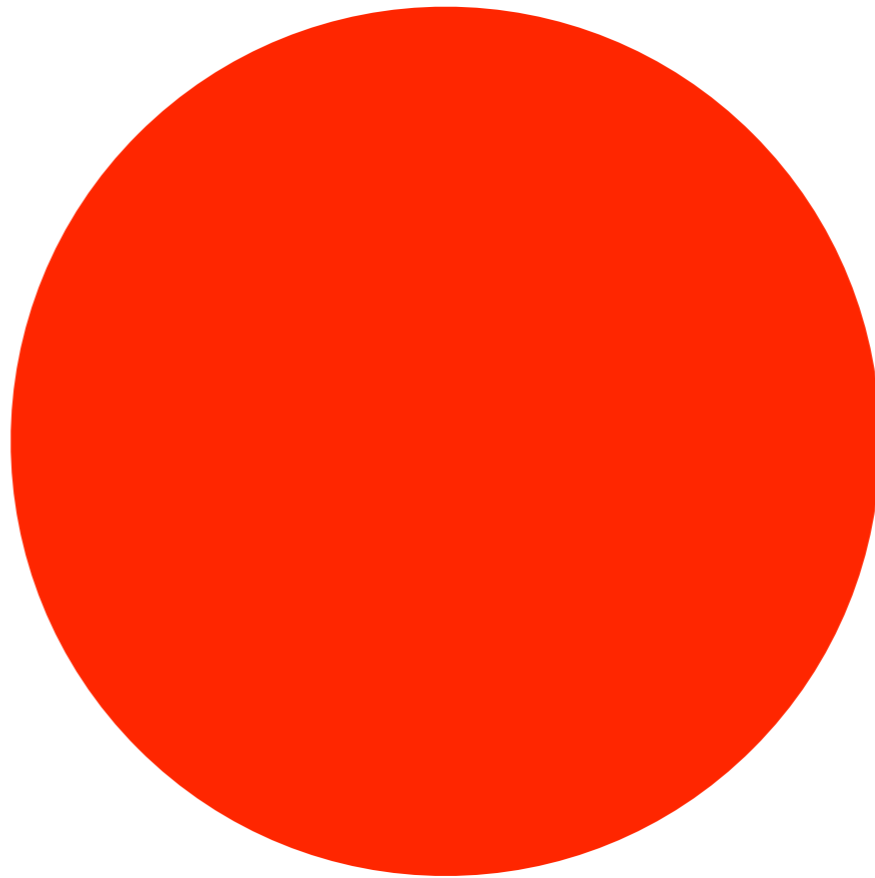
A few things about SVG

- The default starting registration point is in the upper left-hand corner of the `<svg>` box.
- There are some new CSS properties like "fill" and "stroke-width".

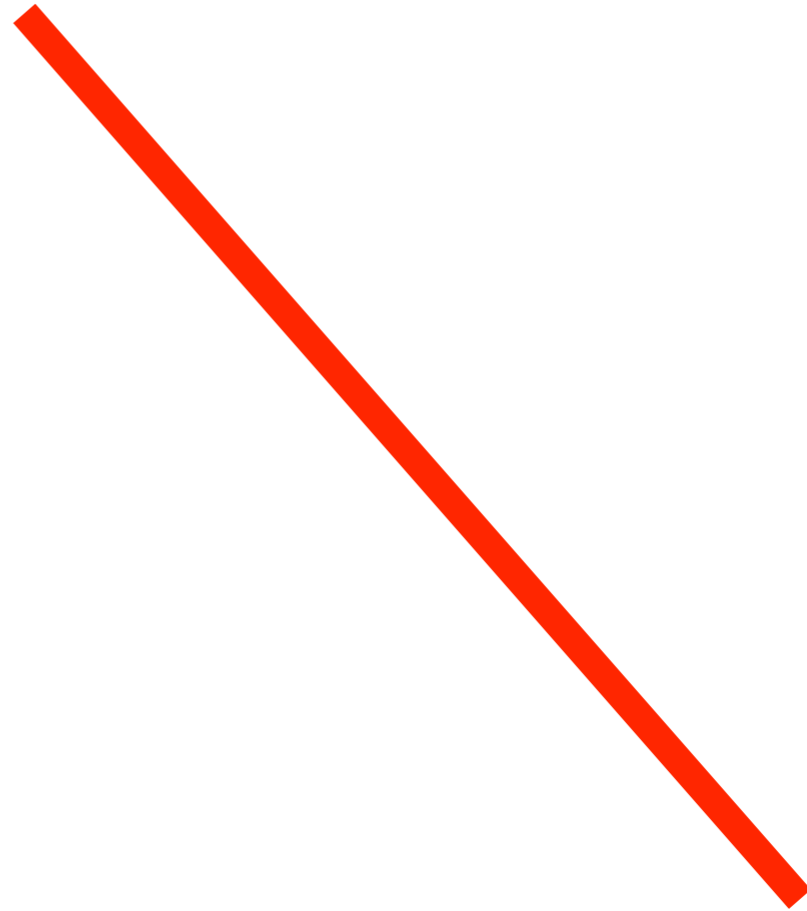
```
<rect  
  width="30"  
  height="10"  
  x="10"  
  y="10">  
</rect>
```



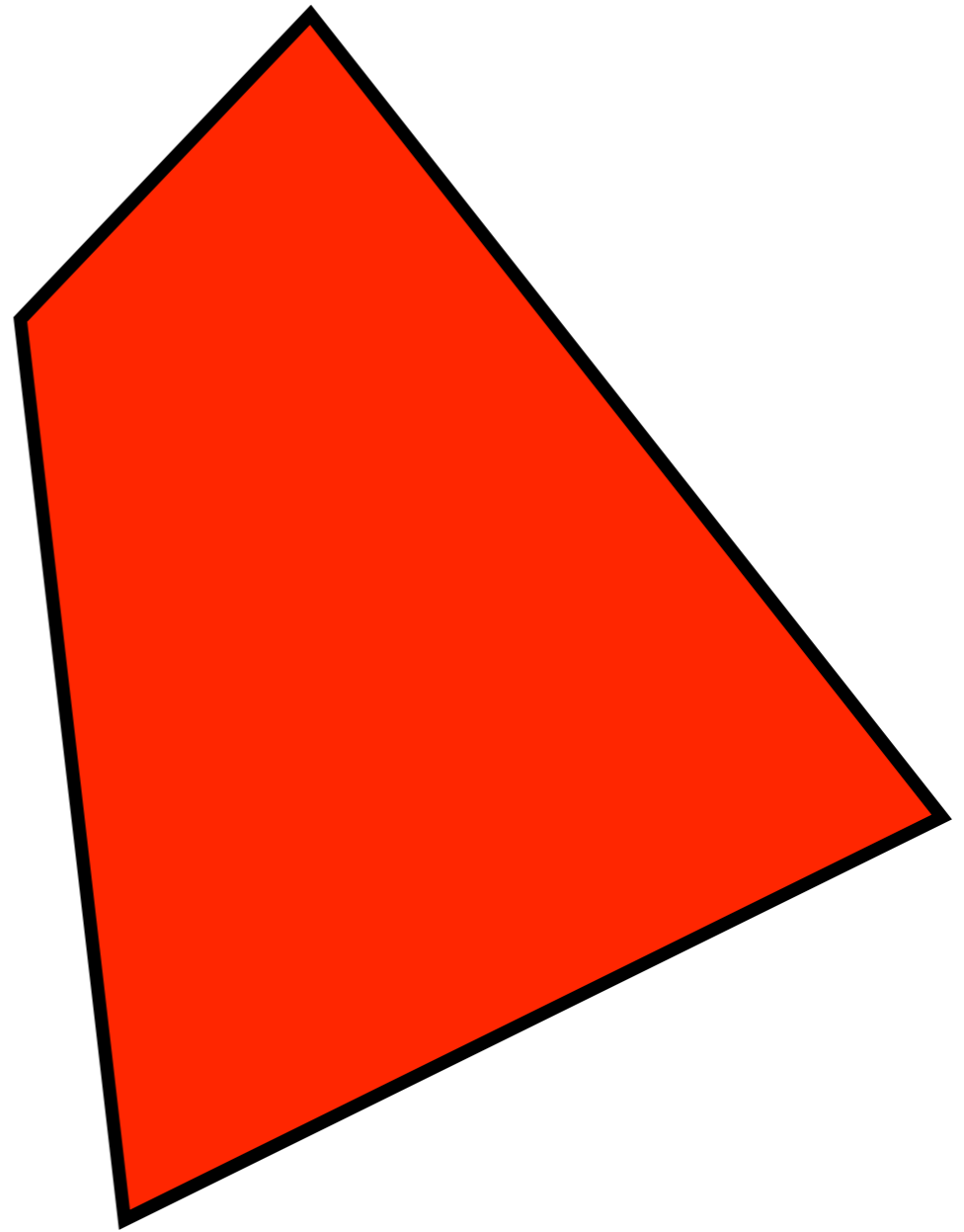
```
<circle  
  cx="0"  
  cy="0"  
  r="10">  
</circle>
```



```
<line  
  x1="0"  
  y1="0"  
  x2="10"  
  y2="10">  
</line>
```



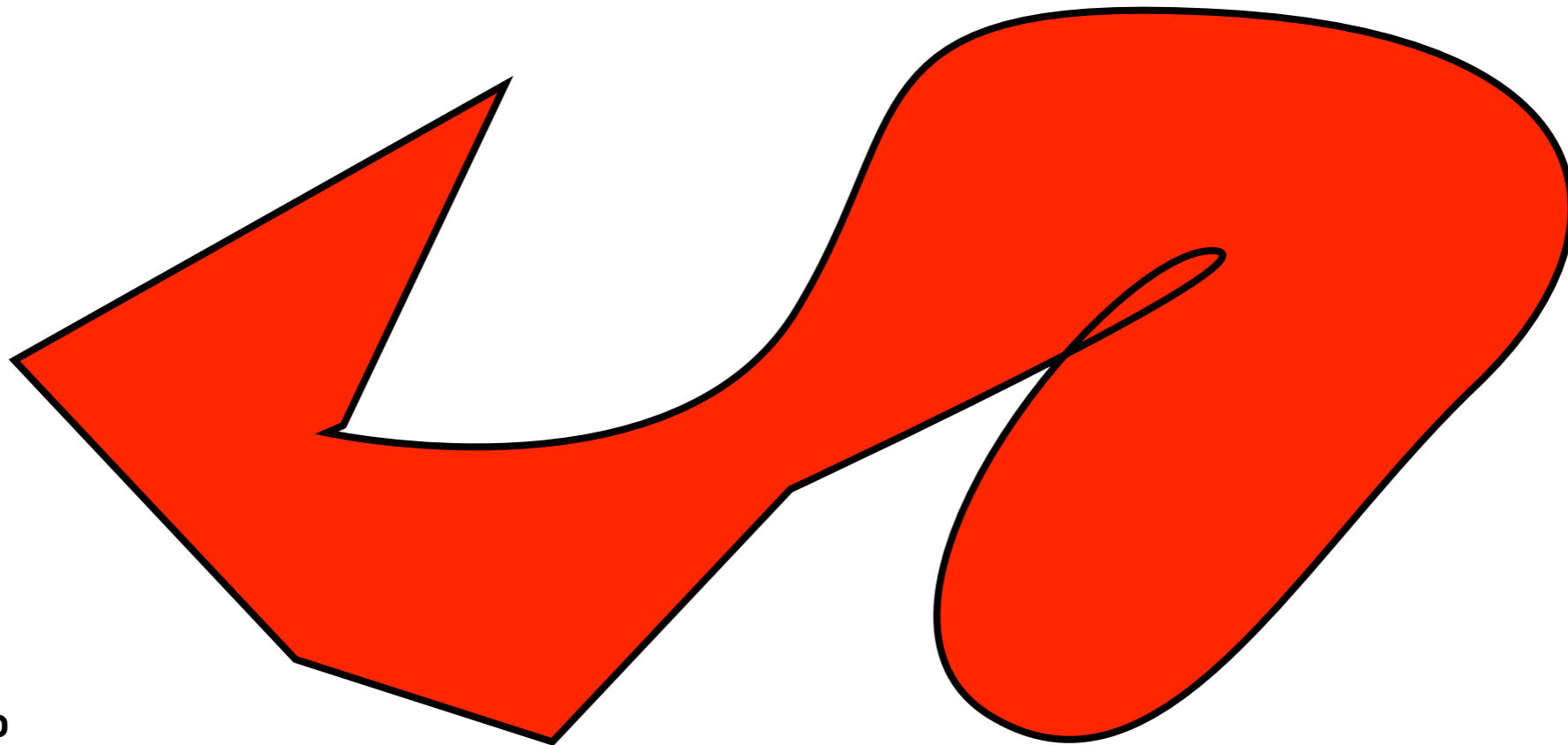

```
<polygon  
  points="  
  10,10 15,5  
  50,50 12,60  
  ">  
</polygon>
```



<path

d="M150 0 L75 200 L225 200 Z">

</path>



- **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**

```
<text
```

```
  x="10"
```

```
  y="10"
```

```
>
```

```
  some text here
```

```
</text>
```

some text

here

```
<tspan  
  x="10"  
  y="10"  
  dx="10"  
  dy="10">  
  some text here  
</text>
```

x and y refer
to document

dx and dy
are relative
(end of
previous line)

<g

x="10"

y="10">

</g>

Transforms

Any tag can have transform attribute

```
<circle transform="" cx="0" cy="0">
```

```
<rect transform="" x="0" y="0">
```

```
<path transform="" d="0,0">
```

```
<polygon transform="" points="0,0">
```

```
<g transform="">
```

Transforms attributes can have
up to three values

```
<circle transform="" cx="0" cy="0">
```



translate(x, y)

rotate(deg, centerX, centerY)

scale(percentage)

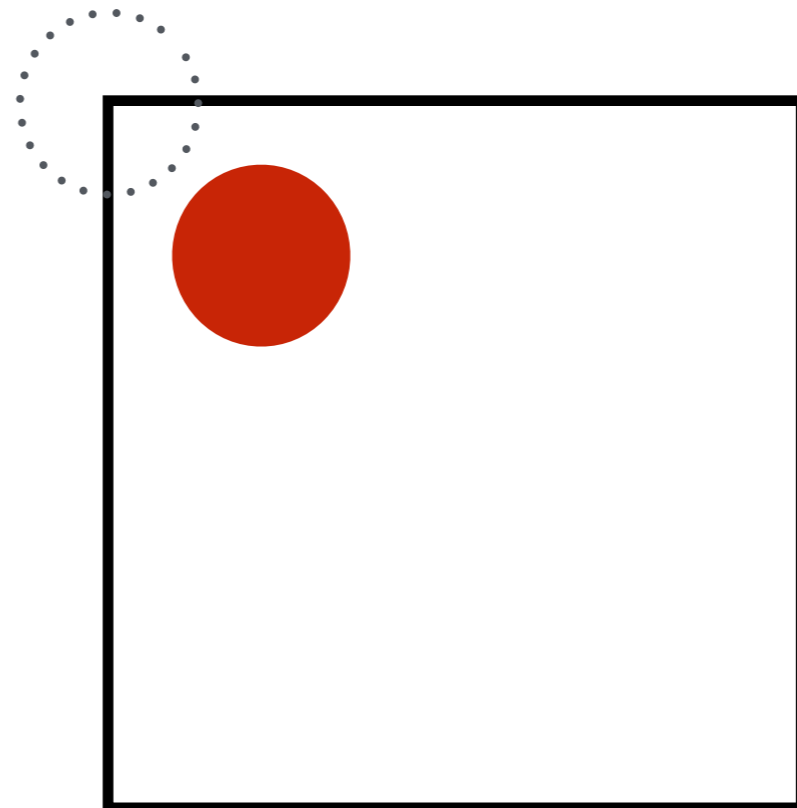
For example...

```
<circle transform="translate(5,5)"
```

```
  cx="0"
```

```
  cy="0"
```

```
  r="10">
```



For example...

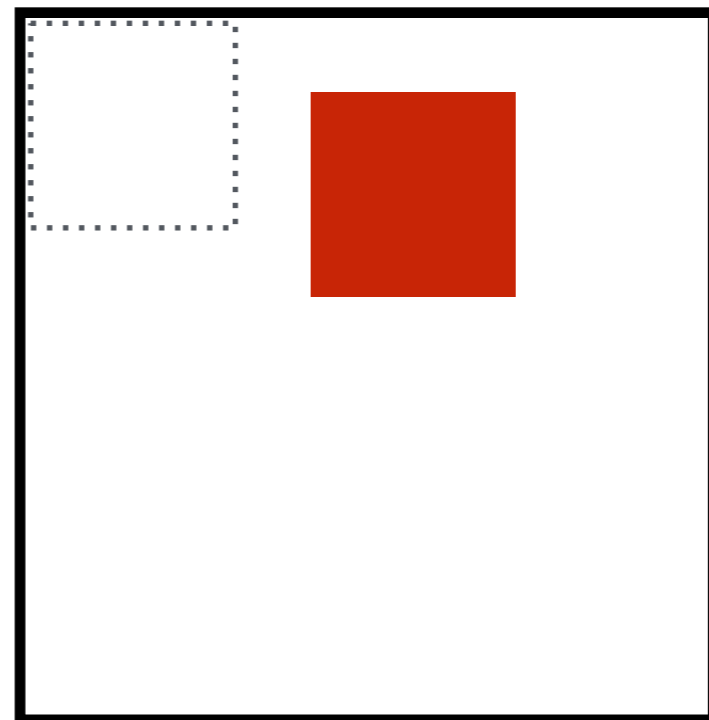
```
<rect transform="translate(50,10)"
```

```
x="0"
```

```
y="0"
```

```
width="20"
```

```
height="20">
```



For example...

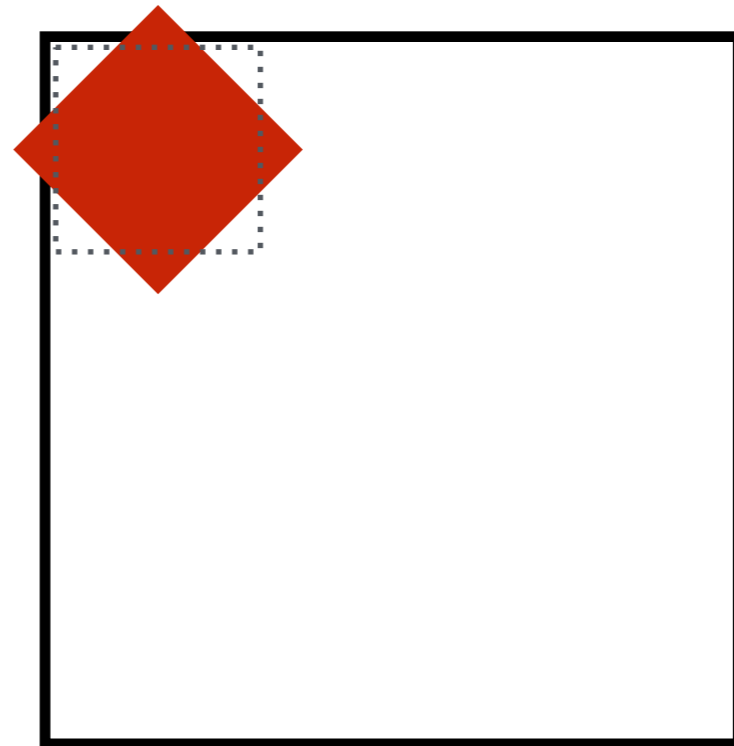
```
<rect transform="rotate(45deg,10,10)"
```

```
x="0"
```

```
y="0"
```

```
width="20"
```

```
height="20">
```



For example...

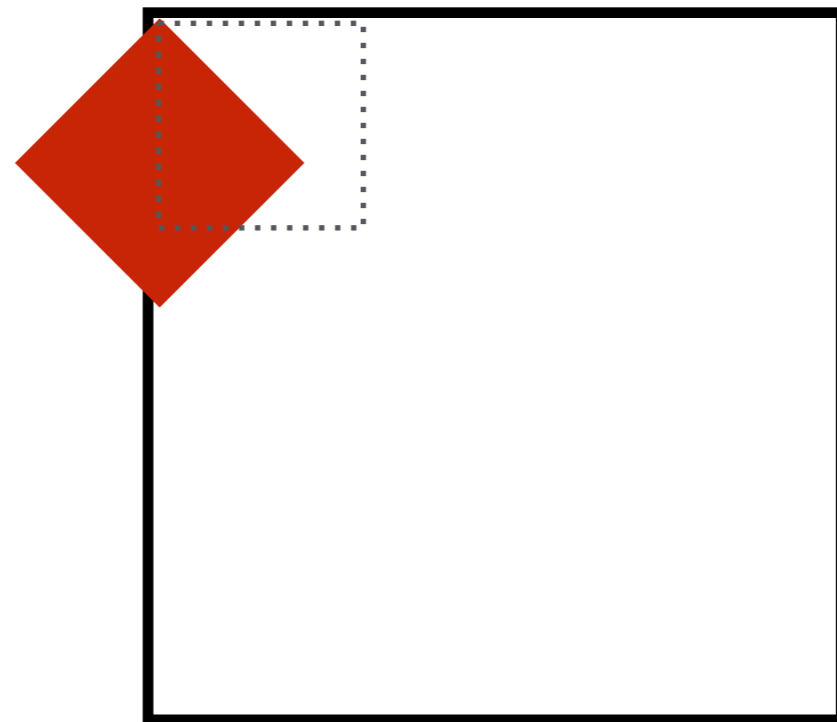
```
<rect transform="rotate(45deg,10,10)"
```

```
x="0"
```

```
y="0"
```

```
width="20"
```

```
height="20">
```



For example...

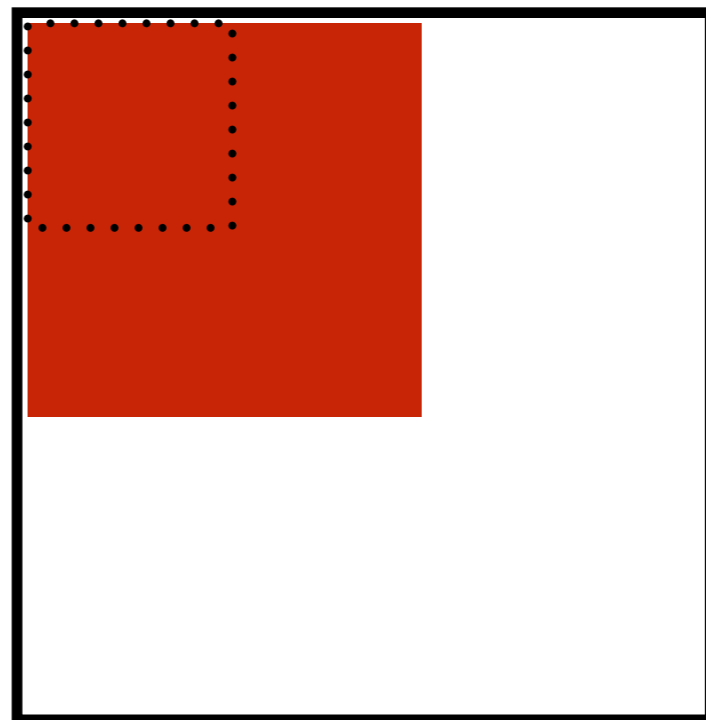
```
<rect transform="scale(2)"
```

```
x="0"
```

```
y="0"
```

```
width="20"
```

```
height="20">
```



For example...

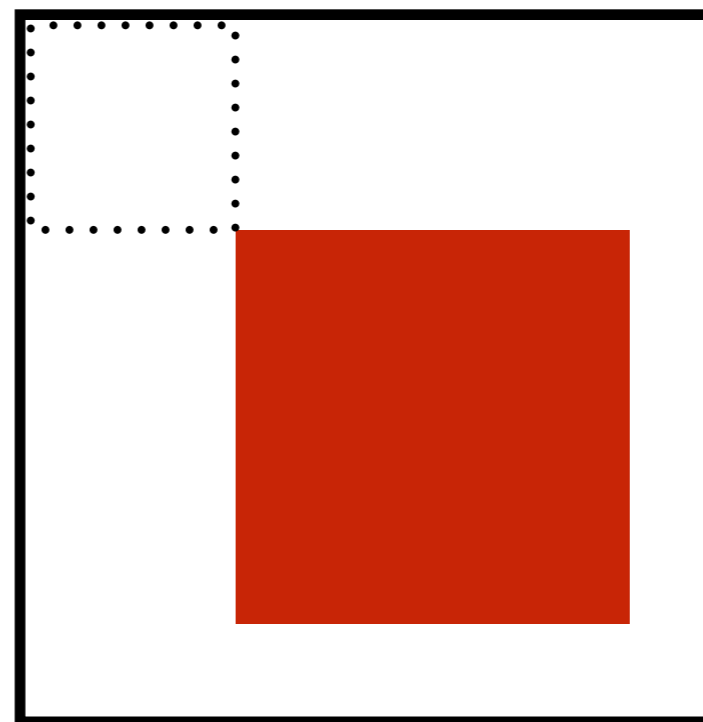
```
<rect transform="scale(2)translate(20,20)"
```

```
x="0"
```

```
y="0"
```

```
width="20"
```

```
height="20">
```



Most common transform

```
<g transform="translate(100,100)">
```

```
  <circle cx="10" cy="10" r="10" fill="red"></circle>
```

```
  <rect x="10" y="10" width="10" height="10" fill="red"></rect>
```

```
  <polygon points="10,10 20,10 15,15" fill="red"></polygon>
```

```
</g>
```

