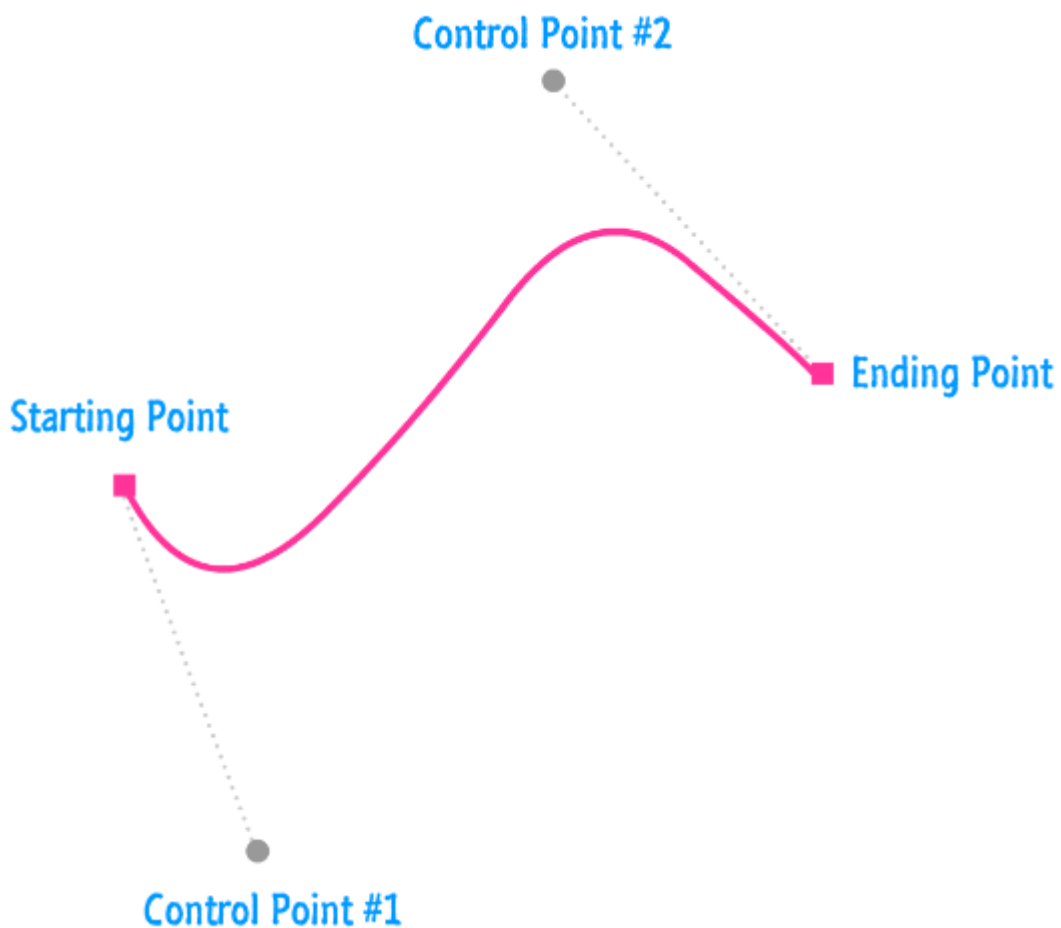


# Drawing a Cubic Beziér Curve

The cubic Beziér curve is almost identical to the quadratic one we just looked at. The only (and very important) difference is that it contains an additional control point:



This additional control point gives your cubic Beziér curve the ability to be smoother and more awesome (such as what you see above) than the quadratic Beziér curve with just a single control point. The way you draw a cubic Beziér curve is by using the `bezierCurveTo` method and specifying the six arguments it needs:

```
context.bezierCurveTo(c1_x, c1_y, c2_x, c2_y, e_x, e_y);
```



The first four arguments specify the coordinates of our two control points. The last two arguments specify the coordinates of our ending point. Just like with our quadratic Beziér curve, the starting point is inferred based on where your virtual pen is prior to calling the `bezierCurveTo` method.

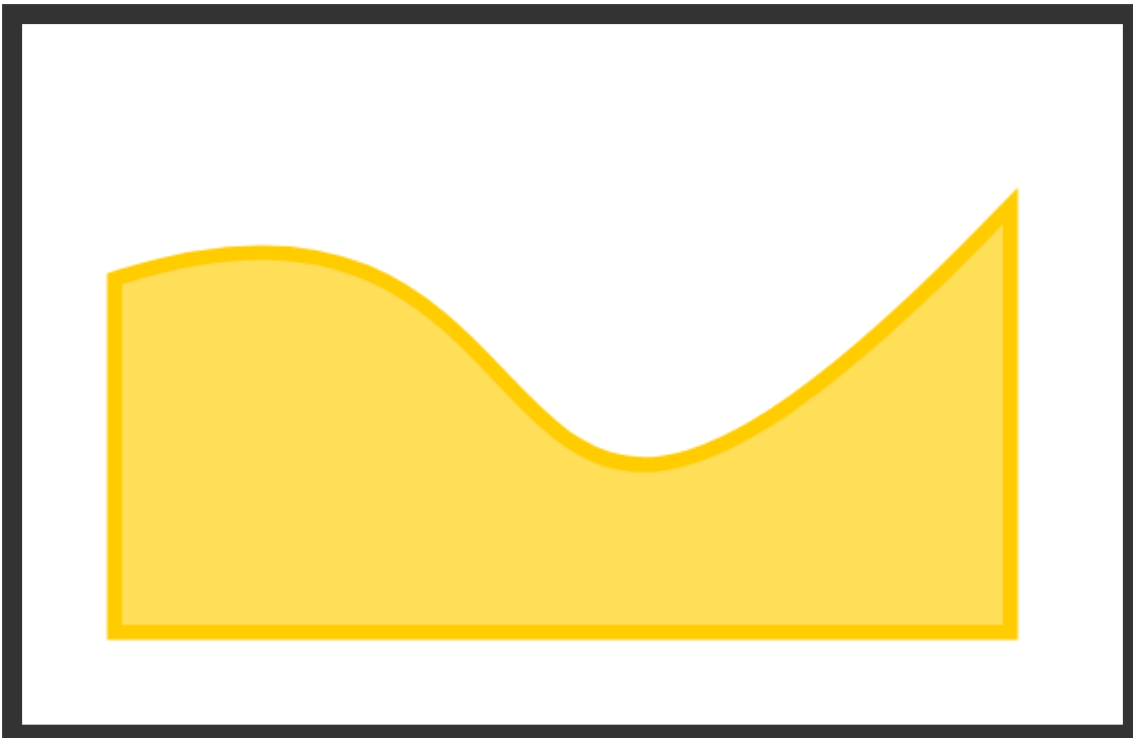
Below is an example that highlights the `bezierCurveTo` method at work:

HTML JavaScript

```
1 var canvas = document.querySelector("#myCanvas");
2 var context = canvas.getContext("2d");
3
4 context.moveTo(50, 130);
5 context.bezierCurveTo(300, 50, 200, 400, 490, 100);
6 context.lineTo(490, 300);
7 context.lineTo(50, 300);
8 context.closePath();
9
10 context.lineWidth = 15;
11 context.strokeStyle = "#FFCC00";
12 context.fillStyle = "#FFDE58";
13
14 context.stroke();
15 context.fill();
```

javascript

output



This example is a little bit more complex than having a single call to the `cubicBezierTo` method, but it does a good job highlighting how well all of the various drawing methods we've looked at so far play together.