

Displaying Our Triangle

Let's start easy by first drawing our triangle. The way we are going to do that is by defining a function called `drawTriangle` that draws a triangle at a fixed position in our `canvas`.

Using our usual example where we have a `canvas` element defined with an `id` value of `myCanvas`, ensure the contents of your `script` tag look as follows:

HTML JavaScript

```
1 var canvas = document.querySelector("#myCanvas");
2 var context = canvas.getContext("2d");
3
4 function drawTriangle() {
5     // the triangle
6     context.beginPath();
7     context.moveTo(200, 100);
8     context.lineTo(170, 150);
9     context.lineTo(230, 150);
10    context.closePath();
11
12    // the outline
13    context.lineWidth = 10;
14    context.strokeStyle = "rgba(102, 102, 102, 1)";
15    context.stroke();
16
17    // the fill color
18    context.fillStyle = "rgba(255, 204, 0, 1)";
19    context.fill();
20 }
21 drawTriangle();
22
```

javascript



Once you have added this code to your document, go ahead and preview your document in your browser. If everything worked out properly, you'll see a yellow triangle displayed. There is nothing exciting going on with this code that you haven't seen before, but there is one thing I want to call out. Notice that our triangle is defined by the following X and Y values:

```
context.moveTo(200, 100);  
context.lineTo(170, 150);  
context.lineTo(230, 150);
```



It is these values that we'll eventually end up adjusting to accommodate our arrow key presses.