Drawing lines with different styles

COMMON MISTAKE: DRAWING THE SAME PATH TWICE

Let's look at the drawing from the last example of the previous section:

Imagine that we would like to draw them with different styles and colors: the shape on the left will stay as it is now (blue, wireframe), while the second shape on the right will be filled, colored in pink. Let's look at how we can do this...

DRAWING TWO PATHS WITH DIFFERENT STYLES, THE WRONG AND THE GOOD WAY!

First, the wrong way!

In this example, we will draw the two parts of the path with different styles: the first part in wireframe mode, and the second part in filled mode.

What we will try first is to call `stroke()` after the first half of the path, then call `fill()` after the second half of the path (check this interactive example):
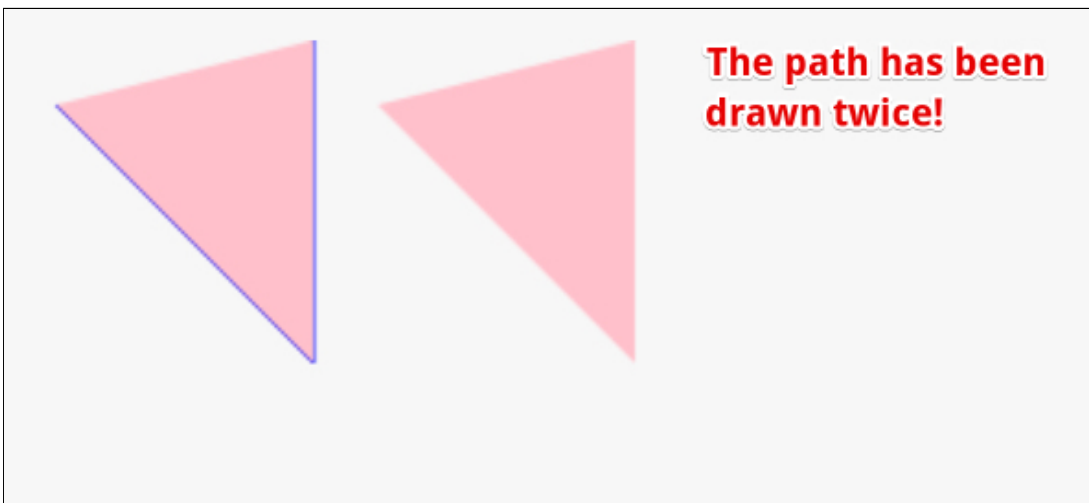
Here is the code:

```
var canvas=document.getElementById('myCanvas');
var ctx=canvas.getContext('2d');
// first part of the path
ctx.moveTo(20,20);
ctx.lineTo(100, 100);
ctx.lineTo(100,0);
// indicate stroke color + draw first part of the path
10. ctx.strokeStyle = "#0000FF";
ctx.stroke();
// second part of the path
ctx.moveTo(120,20);
ctx.lineTo(200, 100);
ctx.lineTo(200,0);
// indicate stroke color + draw the path
ctx.fillStyle = "pink";
20. ctx.fill();
```

Hey - it does not work! Weirdly, the two parts of the path are filled in pink! But we called `stroke()` after the first half of the path was drawn (lines 5-8). Then we called `fill()` only after the second part of the path was specified (lines 14-19)... so, what happened?

Remember that `fill()` or `stroke()` draws the whole path, even if it is disconnected, and even if it has already been drawn!
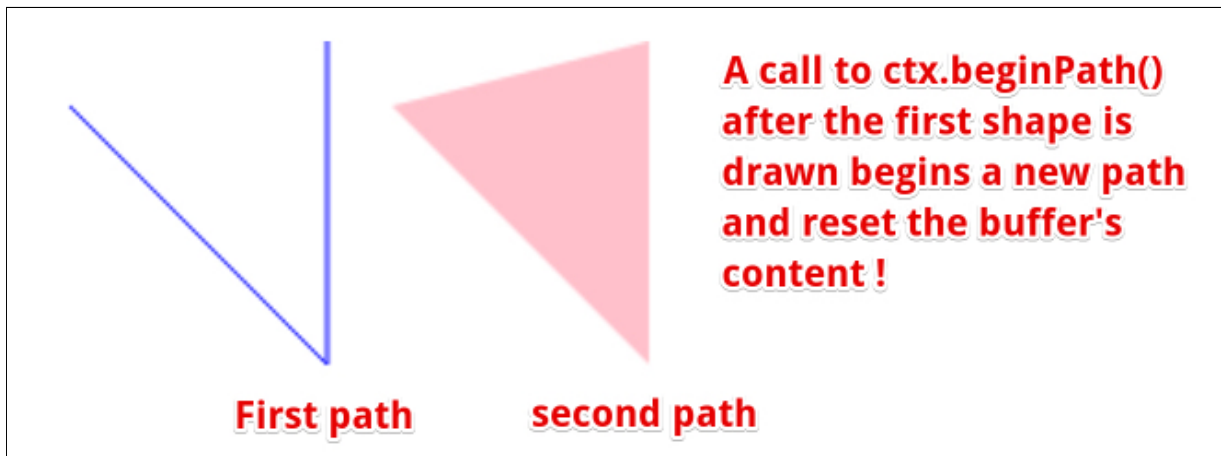
What happened is:

1. The call to `stroke()` has drawn the path corresponding to the lines 5-7. Indeed, the first part of the path (on the left) has actually been drawn once in wireframe mode, and in blue.

2. Then, the call to `fill()` at line 20 has drawn the whole path again, but in pink and in filled mode. But this time the path corresponds to lines 5-7 plus lines 14-16 that make up the second shape on the right. So the path that has been drawn this time is made of both of the triangles.

Important: If you do not want to draw parts of the same path several times, you need to draw two different paths, using the `ctx.beginPath()` method, as shown in the next example.

Now, the right way!

Online example:

A call to ctx.beginPath() after the first shape is drawn begins a new path and reset the buffer's content !

First path   second path

Source code:

```
var canvas=document.getElementById('myCanvas');
var ctx=canvas.getContext('2d');
// first part of the path
ctx.moveTo(20,20);
ctx.lineTo(100, 100);
ctx.lineTo(100,0);
// indicate stroke color + draw first part of the path
10.  ctx.strokeStyle = "#0000FF";
ctx.stroke();
// start a new path, empty the current buffer
ctx.beginPath();
// second part of the path
ctx.moveTo(120,20);
ctx.lineTo(200, 100);
ctx.lineTo(200,0);
20.
// indicate stroke color + draw the path
ctx.fillStyle = "pink";
ctx.fill();
```

This time, in order to draw the two shapes differently, we defined two separate paths. The way to do this is just to call `ctx.beginPath()` to start a new path. In this example, the first path has been drawn in wireframe mode, then a new path has been started that is drawn in filled

mode.

KNOWLEDGE CHECK 3.4.4 (NOT GRADED)

_____

We must call `ctx.beginPath()` before drawing any new path, but what does it do exactly?

- ○ It will reset all properties of the graphic context.

  ○ It will empty the current path (reset the buffer of drawing orders), but it will not change the context properties.