JavaScript Scale

Summary: in this tutorial, you'll learn how to use the JavaScript `scale()` Canvas API to scale drawing objects effectively.

Introduction to JavaScript scale() Canvas API

The `scale()` is a method of the 2D drawing context. The `scale()` method adds a scaling transformation to the canvas units horizontally and/or vertically.

The default unit on the canvas is one pixel. The scaling transformation changes this default.

For example, a scaling factor of 0.5 will change the unit size to 0.5 pixels. Similarly, a scaling factor of 2 will increase the unit size to two pixels.

When the unit size is scaled, shapes are also drawn at the new units.

The following illustrates the `scale()` method:

```
ctx.scale(x, y)
```

The `scale()` method accepts two parameters:

- `x` is the scaling factor in the horizontal direction.
- `y` is the scaling factor in the vertical direction.

JavaScript scale() examples

The following example first draws a rectangle starting at `(10,10)` with a width of 100 pixels and a height of 50 pixels. It then adds a scaling factor of 2 to the canvas both vertically and horizontally. Finally, it draws the rectangle with the same width and height. However, the second rectangle is four times bigger than the first one because of the scaling transformation.

HTML

```html
<canvas id="canvas" height="300" width="500">
</canvas>
```

JavaScript

```javascript
const canvas = document.querySelector('#canvas');

if (canvas.getContext) {

    // rectangle width and height
    const width = 100,
        height = 50;

    // starting point
    const x = 10,
        y = 10;

    const ctx = canvas.getContext('2d');

    ctx.strokeStyle = 'red';
    ctx.strokeRect(x, y, width, height);

    ctx.scale(2, 2);

    ctx.strokeStyle = 'rgba(0,0,255,.2)';
    ctx.strokeRect(x / 2, y / 2, width, height);
}
```
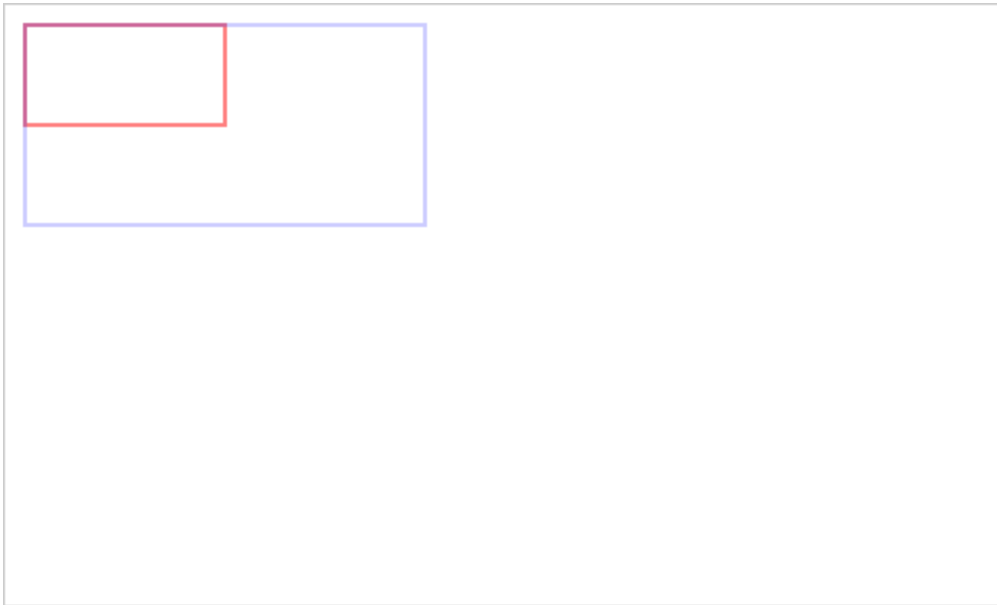
Output:

Here is the demo link.

Summary

- Use the JavaScript `scale()` to add a scaling transformation to the canvas units horizontally and/or vertically