\* {

position: relative;

}

The **relative flow** is a variant of the normal flow and behaves basically the same. But, the relative flow adds some new abilities.

**The relative flow allows you to shift the position of elements *after* they've been laid out in the normal flow.**

.relative {

position: relative;

top: 10px;

left: 10px;

bottom: 10px; /\* redundant if you've already defined top \*/

right: 10px; /\* redundant if you've already defined left \*/

}

You can think of these new CSS properties, top, left, bottom and right, as adjustments to the normal flow. Setting any of them to any pixel value will shift the element by that much from where it *would* have ended up in the normal flow.

[Here is the example](http://udacity.github.io/fend/fend-refresh/lesson6/relative-flow/shift-two.html) from the animation.

Note that you can use positive and negative pixel values. You aren't changing the width or height, only the position. As you can't shift an element both directions in one axis simultaneously, you should only use one of top and bottom or one of left and right at a time.

Let's add another <div> and break down this example some more. Here's the HTML to start (notice that position: relative is commented out):

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Relative Flow Example</title>

<link href='https://fonts.googleapis.com/css?family=Source+Sans+Pro' rel='stylesheet' type='text/css'>

<style>

\* {

box-sizing: border-box;

font-family: 'Source Sans Pro', sans-serif;

font-weight: bold;

}

div {

display: inline-block;

border: 2px solid #2e3d49;

width: 300px;

height: 50px;

}

div:nth-child(even) {

background-color: #02b3e4;

}

div:nth-child(odd) {

background-color: #ecc81a;

}

.shift {

/\*position: relative;\*/

top: 10px;

left: 10px;

}

.relative {

position: relative;

}

</style>

</head>

<body>

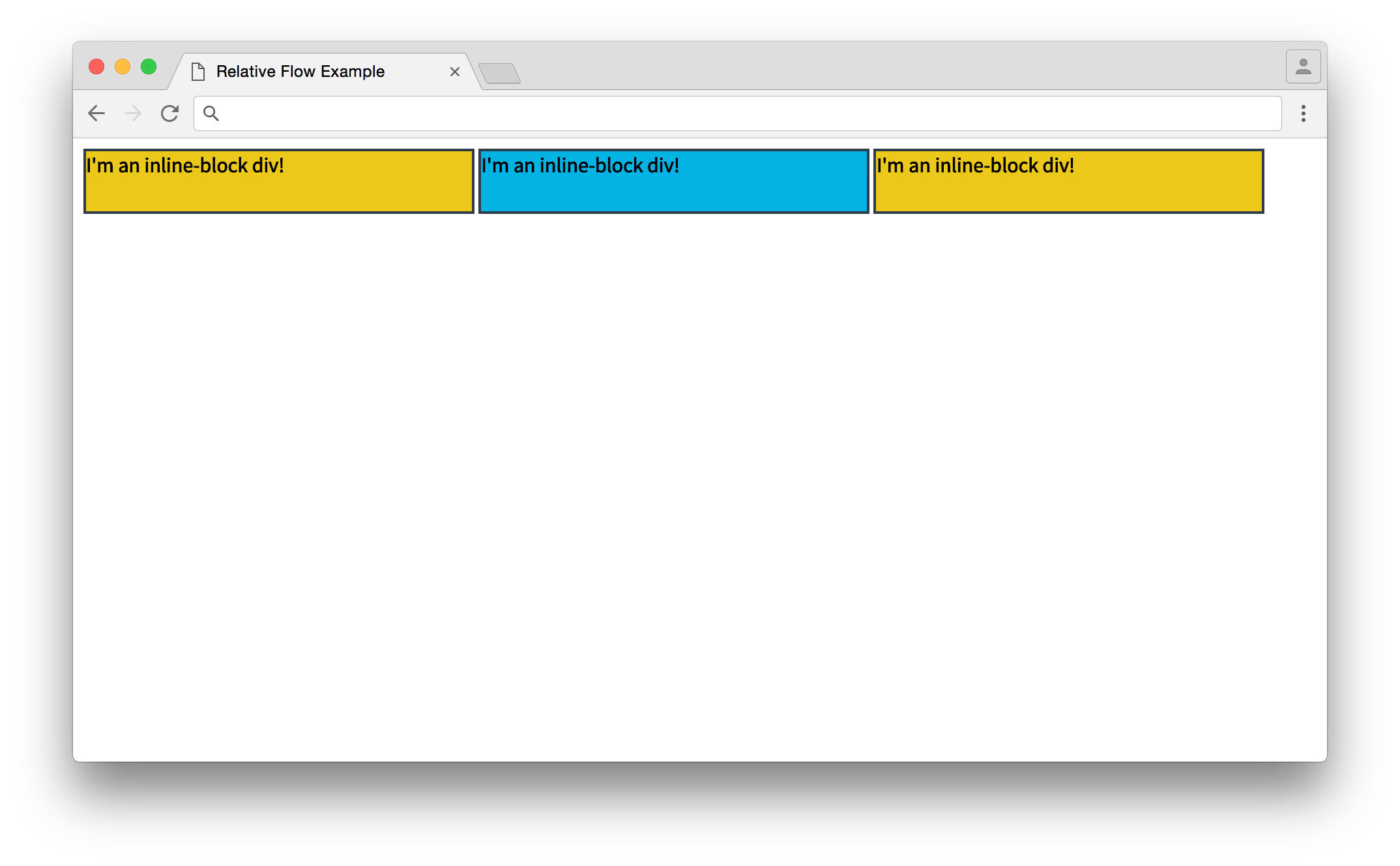
<div>I'm an inline-block div!</div>

<div class="shift">I'm an inline-block div!</div>

<div>I'm an inline-block div!</div>

</body>

</html>

[[](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)

[The normal flow. As all three divs are inline-block, they display in the same line.](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)

[This shows](http://udacity.github.io/fend/fend-refresh/lesson6/relative-flow/shift-three.html) exactly what you'd expect with the normal flow. Now I'll uncomment position: relative in the .shift ruleset, which should change the position of the second <div> (you should try this out with developer tools on your own!).

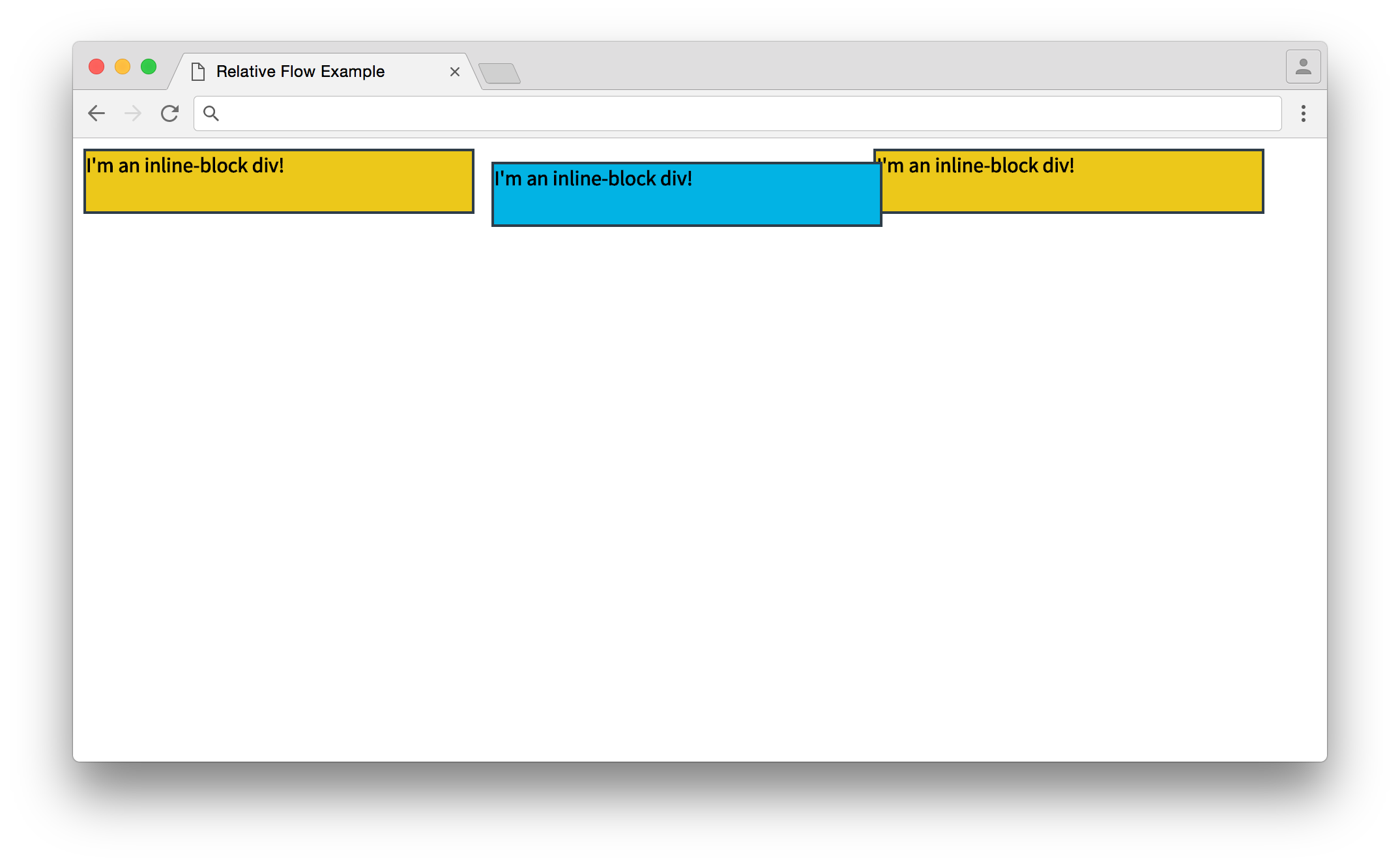
.shift {

position: relative;

top: 10px;

left: 10px;

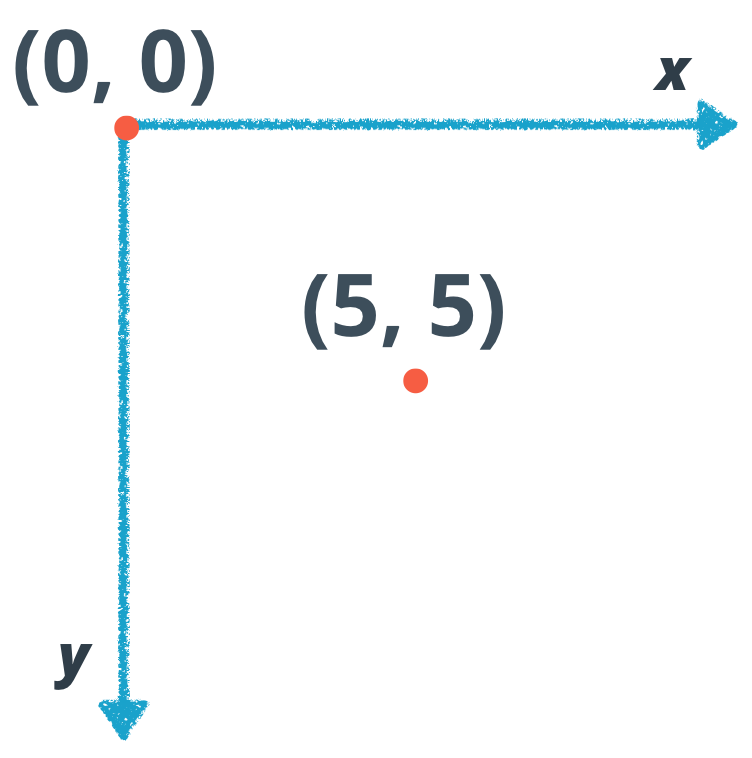
}

[[](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)

[The relative flow. The second div is in a new position!](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)

The relatively positioned .shift element moved from its original position down by 10 pixels and right by 10 pixels. Now, it may seem weird that the element moved *down* when the top property was added, but it makes sense when you consider the [coordinate system](https://en.wikipedia.org/wiki/Coordinate_system) that browsers use.

You're probably familiar with the [Cartesian coordinate system](https://en.wikipedia.org/wiki/Cartesian_coordinate_system) from geometry or the [Polar coordinate system](https://en.wikipedia.org/wiki/Polar_coordinate_system) from trigonometry. Browsers use a slightly different coordinate system.

[[](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)

[The coordinate system that browsers use. The vertical axis, y, starts at 0 at the top and gets more positive as you go down, while the horizontal axis, x, starts at 0 on the left and gets more positive as you move right.](https://classroom.udacity.com/nanodegrees/nd004-intr1/parts/4755cfe6-bd20-46a8-afc9-4c513d17d72f/modules/c1039f15-30d9-4fef-a276-088c5ae93af1/lessons/5bc1143d-6294-412b-9a32-27d17142f9d2/concepts/3346831f-e58c-4aca-863c-a785d6bc56f7)

Similar to the Cartesian coordinate system, there are *x* and *y* axes. But while the *y*-axis in the Cartesian system gets more positive as you go *up*, the *y*-axis in browsers starts with 0 at the top and gets more positive as you move *down*.

Thus, when you set top to a positive number, you're actually moving the element down. And if you make left a positive number, you move the element to the right!

Did you notice how the third element didn't move in the last example? Even after the second element shifted, the third element stayed in place. You'll be exploring why in the next quiz.