Start/Stop an Animation

In this lesson, you will learn how to start and stop an animation.

WE'LL COVER THE FOLLOWING

Start an Animation

The example code defines a function called `moveBlock()` which moves the block horizontally to the right. It grabs the current position of the block's left border than adds the value contained in the `movement` variable. Next, the code calls the `requestAnimationFrame()` method to keep the animation going.

Position values are written in pixels. These are the strings you saw that resemble "XXpx," which requires the use of the JavaScript `parseFloat()` function to convert numeric values before making calculations.

> Don't use `Number()` to convert a string with `"px"` into a numerical value. This won't work, and you'll get a `NaN` value (Not a Number) as a result!

The `requestAnimationFrame()` function lets you ask the browser to execute a function as soon as possible, which updates the animation. It's the browser's job to make the animation as smooth as possible. The returned value of `requestAnimationFrame()` is an ID for the animation, which can be used to further modify it.

Here is how `requestAnimationFrame()` is used in combination with an animation function.

JavaScript

```
const animate = () => {
    // Animation code
    // ...
    // At end of animation, request another one
    animationId = requestAnimationFrame(animate);
};

// Animation start
let animationId = requestAnimationFrame(animate);
```

Stop an Animation

Let’s now see how to stop the block before it reaches the border of the frame that contains it. We’ll have to verify that the left border position is less than the width of the frame, bearing in mind the thickness of the block itself.

Here’s the updated JavaScript code.

Output

JavaScript

HTML

CSS (SCSS)

```
// Move the block to the right, all the way to the end of the frame
const moveBlock = () => {
  // Convert the left position of the block (value of the form "XXpx") to a number
  const xBlock = parseFloat(getComputedStyle(blockElement).left);
  // Convert the width of the frame (value of the form "XXpx") to a number
  const xMax = parseFloat(getComputedStyle(frame).width);
  // If the block isn't already to the end of the frame
  if (xBlock + blockWidth <= xMax) {
    // Block movement
    blockElement.style.left = (xBlock + movement) + "px";
    animationId = requestAnimationFrame(moveBlock);
  }
  else {
    // Cancel the animation
    cancelAnimationFrame(animationId);
  }
};

const blockElement = document.getElementById("block");
// Convert the block width (value of the form "XXpx") to a number
const blockWidth = parseFloat(getComputedStyle(block).width);

// Movement value in pixels
const movement = 7;
```

```
// Start the animation
let animationId = requestAnimationFrame(moveBlock);
```

The new `moveBlock()` function checks that the block has arrived at the end of the frame before moving. If that’s the case, the animation stops via a call to `cancelAnimationFrame()`. The `cancelAnimationFrame()` functions stops the animation and takes the ID of the animation set by a prior call to `requestAnimationFrame()`.

JavaScript

```
// Stop an animation
cancelAnimationFrame(animationID);
```