

The Timestamp Argument

The last thing we are going to look at is the argument `requestAnimationFrame` passes in to the callback function. The following is the callback function signature I showed you earlier:

```
function draw() {  
  .  
  .  
  .  
  requestAnimationFrame(draw);  
}
```



That is still accurate and totally works. There is an **optional** argument representing the exact time the callback function is called (aka a timestamp) that I kinda did not cover, and a big thanks to [Jamie Perkins \(@inorganik\)](#) for pointing that out!

Below is the slightly different callback function signature with the timestamp argument specified:

```
function draw(timestamp) {  
  .  
  .  
  .  
  requestAnimationFrame(draw);  
}
```



You can use any name for this argument (I went with `timestamp`), and also notice that I didn't change anything about how I call this callback function. It is still `requestAnimationFrame(draw)`.

Anyway, what this timestamp argument represents is a time value accurate to a thousandth of a millisecond for when the callback function was called. If you inspect the value of the timestamp, everytime your callback function gets called, you'll see time values that look similar to the following appear:

```
.  
.   
.   
1390549443371.2961  
1390549443442.2961  
1390549443453.2961  
1390549443489.2961  
1390549443503.2961  
1390549443520.2961  
1390549443536.2961  
1390549443554.296  
1390549443569.2961  
1390549443586.2961  
1390549443604.2961  
1390549443619.2961  
1390549443636.2961  
1390549443653.2961  
1390549443670.2961  
1390549443686.2961  
1390549443704.2961  
1390549443719.2961  
1390549443736.2961  
.   
.   
.
```

At this point, this may not make much sense, and you may also be wondering why this is important. The simple answer is that you have a really high precision counter that you can use to keep track of how many times your callback function was called. To fully see the importance of this, I will explain in greater detail in a future article. In the meantime, the [Animating with Robert Penner's Easing Functions](#) tutorial and the [Incrementer vs. Timestamp](#) examples may provide some clues.