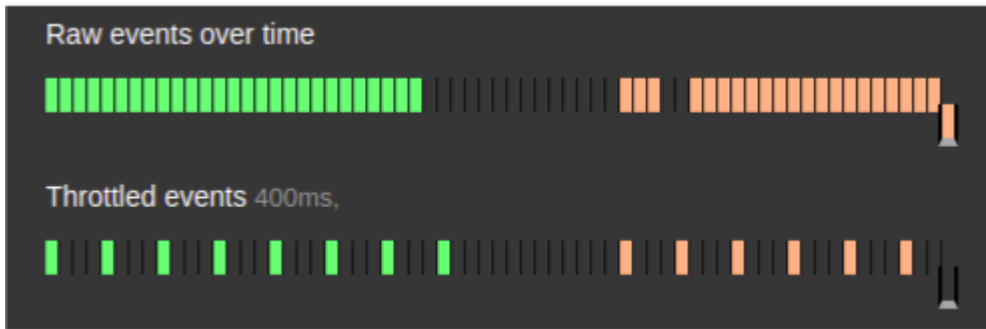


Throttle



Source: <https://css-tricks.com/debouncing-throttling-explained-examples/>

1. Throttle usages
2. Simple implementation
3. debounce and throttle

1. Throttle usages

- **throttle**: là một kỹ thuật để giảm tần suất của các hàm được gọi liên tục, giới hạn chúng chỉ được gọi một lần sau một khoảng thời gian nhất định.
- Điều này thường được sử dụng để kiểm soát tần suất của các sự kiện như **scroll**, **resize** hoặc **keystroke** trong ứng dụng web.

<https://webdesign.tutsplus.com/javascript-debounce-and-throttle--cms-36783t>

2. Simple implementation

```
function log(){
  console.log('Hello World')
}

function throttle(callback, wait) {
  let isThrottling = false;
  return function () {
    if (isThrottling) return; // Nếu đang throttle, return -> không thực hiện
    callback
    isThrottling = true; // Đặt cờ throttle thành true
    setTimeout(() => {
      callback(); // Thực hiện hàm callback
      isThrottling = false; // Đặt lại cờ throttle sau thời gian chỉ định
    }, wait);
  };
}

const throttledLog = throttle(log, 500); // Tạo phiên bản bị throttle của hàm log
với khoảng thời gian 500ms
```

```
// Lần gọi đầu tiên, thực hiện ngay lập tức
throttledLog(); // Thực hiện callback, đặt throttle, lên lịch thực hiện tiếp theo
sau 500ms

// Lần gọi thứ hai, thứ ba và thứ tư, bị bỏ qua do bị throttle
throttledLog(); // Bị throttle, không thực hiện callback
throttledLog(); // Bị throttle, không thực hiện callback
throttledLog(); // Bị throttle, không thực hiện callback

// Sau 500ms, throttle được bỏ qua, và callback có thể được thực hiện lại
// Thực thi sau 600ms (100ms sau khi reset throttle)
setTimeout(throttledLog, 600); // Thực hiện callback, đặt throttle, lên lịch thực
hiện tiếp theo sau 500ms
setTimeout(throttledLog, 600); // Bị throttle
setTimeout(throttledLog, 600); // Bị throttle
setTimeout(throttledLog, 600); // Bị throttle

// Sau 700ms, throttle vẫn đang hoạt động, vì vậy callback không thực hiện ngay
lập tức
setTimeout(throttledLog, 700); // Bị throttle
setTimeout(throttledLog, 700); // Bị throttle
```

3. debounce and throttle

#	debounce	throttle
concept	không giới hạn số lần gọi lại hàm, mà chỉ xác định thời gian trôi qua trước khi hàm được gọi.	giới hạn tần suất gọi hàm chỉ một lần sau một khoảng thời gian nhất định.
reduce number of calls	YES	YES
common case	search when stop typing	do something on scroll