# 每日一题 - 2019-11-25 - 写一个debounce的装饰器

### 信息卡片

* 时间：2019-11-25
* tag：编程题

### 问题描述

实现一个debounce装饰器

### 参考实现

##### 代码实现

/\*\*  
 \* 装饰器的debounce  
 \* @param delay  
 \*/  
export function debounce(delay: number): Function {  
 return (  
 target: Function,  
 propertyKey: string,  
 propertyDesciptor: PropertyDescriptor  
 ) => {  
 const method = propertyDesciptor.value;  
 let timer = null;  
 propertyDesciptor.value = (...args) => {  
 if (timer) {  
 clearTimeout(timer);  
 timer = null;  
 }  
 timer = setTimeout(() => method(...args), delay);  
 };  
 return propertyDesciptor;  
 };  
}

##### 单元测试

import { debounce } from './index';  
  
jest.useFakeTimers();  
  
let a: any;  
let mockFunc: jest.Mock;  
beforeEach(() => {  
 mockFunc = jest.fn();  
 class Test {  
 @debounce(1000)  
 sayHi() {  
 mockFunc();  
 }  
 }  
 a = new Test();  
});  
  
describe('debounce:', () => {  
 test('debounced function should be called after the delay time', () => {  
 a.sayHi();  
 expect(mockFunc).toHaveBeenCalledTimes(0);  
 jest.advanceTimersByTime(1000);  
 expect(mockFunc).toHaveBeenCalledTimes(1);  
 });  
  
 test('debounced function should not be called before the delay time', () => {  
 a.sayHi();  
 expect(mockFunc).toHaveBeenCalledTimes(0);  
 let count = 100;  
 while (count--) {  
 a.sayHi();  
 }  
 expect(mockFunc).toHaveBeenCalledTimes(0);  
  
 count = 100;  
 while (count--) {  
 jest.advanceTimersByTime(999);  
 a.sayHi();  
 }  
 expect(mockFunc).toHaveBeenCalledTimes(0);  
 });  
});

执行结果

### 扩展

* 写一个throttle的装饰器