

Sobre decorators

Gerson implementou o decorator visto neste capítulo. Contudo, ao usar em métodos que precisavam realizar `event.preventDefault()`, ele não funcionou corretamente, muitas vezes recarregando a página, o que não faz sentido dentro de uma Single Page Application.

Seu código com problema esta assim:

```
export function throttle(milisse segundos = 500) {  
  
  return function(target: any, propertyKey: string, descriptor: PropertyDescriptor)  
  
    const metodoOriginal = descriptor.value;  
  
    if(event) event.preventDefault();  
  
    descriptor.value = function(...args: any[]) {  
      let timer = 0;  
      clearInterval(timer);  
      timer = setTimeout(() => metodoOriginal.apply(this, args), milisse segundos);  
    }  
  
    return descriptor;  
  }  
}
```

Marque apenas as opções verdadeiras que indiquem problemas no código escrito por Gerson.

Selecione 2 alternativas

- A** A declaração de `let timer` não pode estar dentro da função passada para `descriptor.value`, caso contrário toda vez que o método decorador for chamado uma nova variável será criada, queremos atualizar a mesma em cada chamada, por isso ela deve ficar no escopo de `throttle` e não do `descriptor`.
- B** Ele não deveria retornar o `descriptor`, já que o está modificando.
- C** A instrução `if(event) event.preventDefault();` esta fora do bloco da função passada para `descriptor.value`.