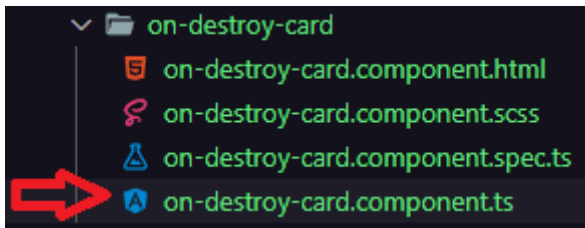


## OnDestroy

Esse método é chamado antes de um componente ser destruído. É usado para executar qualquer lógica de limpeza necessária, como cancelar assinaturas de observáveis, desligar temporizadores ou liberar recursos.

Para Utilizar:

1) Para utilizar o método “OnDestroy” precisamos importa-lo e implementa-lo na classe do arquivo “.ts” do componente.

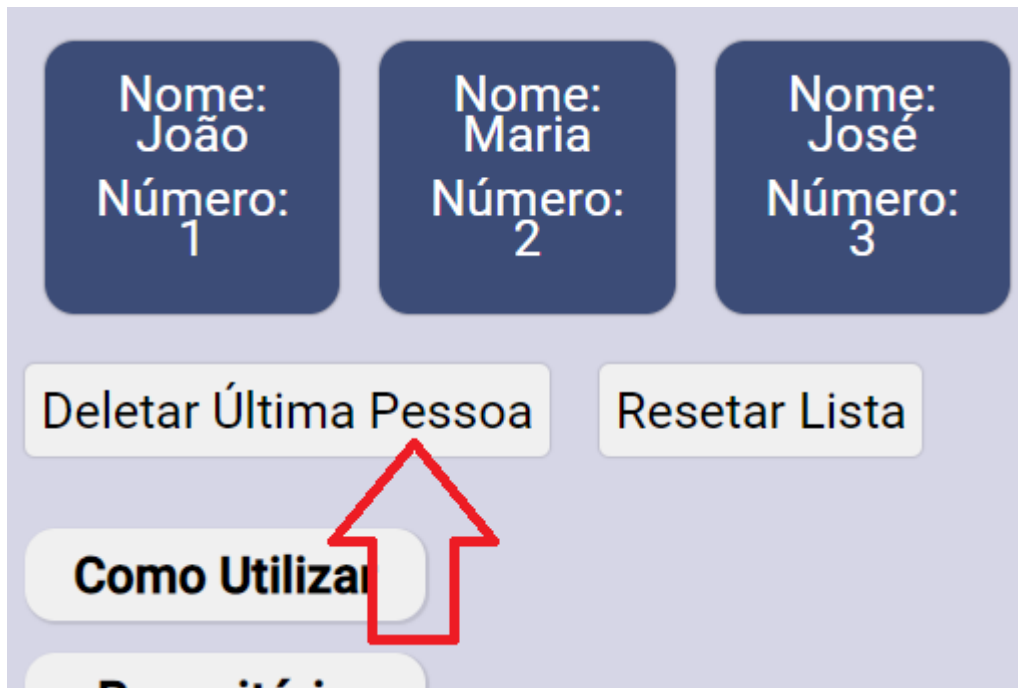


```
1 import { Component, Input, OnDestroy } from '@angular/core';
2
3 @Component({
4   selector: 'app-on-destroy-card',
5   templateUrl: './on-destroy-card.component.html',
6   styleUrls: ['./on-destroy-card.component.scss']
7 })
8 export class OnDestroyCardComponent implements OnDestroy{
9
10   @Input() nome: string = ''
11   @Input() numero:string = ''
12
13   ngOnDestroy(): void {
14     console.log('card deletado - OnDestroy');
15   }
16 }
17
18
19
```

2) Agora podemos configurar o método utilizando a chamada como “ngOnDestroy” e dizer o que será executado.

```
1 import { Component, Input, OnDestroy } from '@angular/core';
2
3 @Component({
4   selector: 'app-on-destroy-card',
5   templateUrl: './on-destroy-card.component.html',
6   styleUrls: ['./on-destroy-card.component.scss']
7 })
8 export class OnDestroyCardComponent implements OnDestroy{
9
10   @Input() nome: string = ''
11   @Input() numero:string = ''
12
13   ngOnDestroy(): void {
14     console.log('card deletado - OnDestroy');
15   }
16 }
17
18
19
```

3) Desta forma, sempre que o componente for removido, será impresso no console do navegador a mensagem "card deletado - OnDestroy"



Esse método é chamado antes de um componente ser destruído. É usado para executar qualquer lógica de limpeza necessária, como cancelar assinaturas de observáveis, desligar temporizadores ou liberar recursos. Neste exemplo, quando removemos um item da lista, o componente filho "on-destroy-card" executa a ação do método "OnDestroy" e imprime no console.

