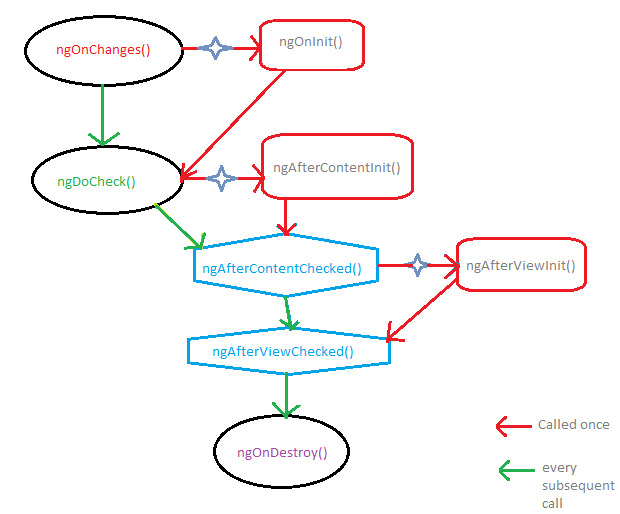
Report

Angular tutorial for beginners  
(20/07/2023)

# **Angular life cycle hook:**



## **constructor:**

* Được gọi một lần duy nhất khi một instance của lớp đó được tạo ra constructor
* Thường được sử dụng để khởi tạo các thuộc tính của lớp và thực hiện các thao tác khởi tạo cần thiết.
* Contructor còn được sử dụng để tiêm các dependency vào component.

## **ngOnChange:**

* Được gọi mỗi khi có sự thay đổi giá trị từ các @Input của component. (nếu component không có input thì hàm này không được gọi đến)
* Phương thức ngOnChanges() nhận vào một đối tượng SimpleChanges chứa thông tin về các thay đổi của các Input. changes[‘@Input’]. previousValue/ currentValue

## **ngOnInit:**

* Chỉ được gọi một lần, sau ngOnChanges() đầu tiên.
* Hook này chỉ ra rằng Angular đã hoàn tất việc tạo ra component.
* Sử dụng ngOnInit để làm một số việc sau khi component đã được tạo ra:
  + Khai báo biến
  + Gọi các Function

## **ngDoCheck:**

* Được gọi mỗi khi change detector của component được gọi đến sau hàm ngOnChanges và một lần duy nhất sau ngOnInit.
* Hay có thể nói nó được gọi bởi Angular sau mỗi lần chạy kiểm tra thay đổi (change detection).
* Sử dụng ngDoCheck để kiểm tra xem một thuộc tính cụ thể có thay đổi hay không và thực hiện các hành động cần thiết khi có sự thay đổi.
* Lưu ý rằng việc sử dụng ngDoCheck có thể ảnh hưởng đến hiệu suất của ứng dụng, do đó cần cân nhắc kỹ trước khi sử dụng hook này.

## **ngAfterContentInit :**

* Gọi duy nhất 1 lần sau hàm ngDoCheck chạy lần đầu tiên. Phản hồi đối với sự kiện content bên ngoài được project (chèn?) vào template của component.

## **ngAfterContentChecked:**

* Được gọi sau hàm ngAfterContentInit và sau mỗi hàm ngDoCheck lần thứ 2 trở đi. Phản hồi sau khi Angular check xong content bên ngoài đã được project vào component template.

## **ngAfterViewInit:**

* Được gọi 1 lần duy nhất sau hàm ngAfterContentChecked chạy lần đầu tiên. Phản hồi sau khi các view của component đã được Angular khởi tạo xong.

## **ngAfterViewChecked:**

* Gọi sau hàm ngAfterViewInit và các lần chạy hàm ngAfterContentChecked lần thứ 2 trở đi. Phản hồi sau khi Angular đã check xong các view của component.

## **ngOnDestroy:**

* Được gọi trước khi huỷ bỏ component. Có thể đặt các lệnh dọn dẹp để giải phóng tài nguyên
  + unsubscribe observable và DOM events
  + Stop interval timers
  + Unregister tất cả callbacks mà directive đã đăng ký với global hoặc application services

# **RxJS:**

## **Observable:**

## **Observer:**

## **Operators:**

### **Marble diagrams:**

Marble diagrams là biểu diễn trực quan về cách thức hoạt động của các operator và bao gồm (các) input Observable, operator và các parameters của chúng cũng như output Observable.

Trong marble diagrams, thời gian chạy về bên phải và sơ đồ mô tả cách các giá trị ("marbles") được phát ra trong quá trình thực thi observable.

### **Categories of operators:**

<https://rxjs.dev/guide/operators#categories-of-operators>

## **Creating custom operators**

### **Use the pipe() function:**

### **Creating new operators from scratch:**

## **Subscription:**

Subscription là một đối tượng đại diện cho một tài nguyên dùng một lần, thường là việc thực thi một Observable. Subscription có một phương thức quan trọng, unsubscribe, không cần đối số và chỉ xử lý tài nguyên do subscription nắm giữ.

Về cơ bản, subscription chỉ có chức năng unsubscribe() để giải phóng tài nguyên hoặc hủy thực thi Observable.

## **Subject:**

Subject RxJS là một loại Observable đặc biệt cho phép các giá trị được multicasted tới Observers.